



REPLICATIVE USE OF AN EXTERNAL MODEL  
IN SIMULATION VARIANCE REDUCTION

THESIS

Thomas H. Irish  
Captain, USAF

AFIT/GOA/ENS/96M-02

**DISTRIBUTION STATEMENT A**

Approved for public release;  
Distribution Unlimited

DEPARTMENT OF THE AIR FORCE  
AIR UNIVERSITY  
**AIR FORCE INSTITUTE OF TECHNOLOGY**

Wright-Patterson Air Force Base, Ohio

19970501 176

AFIT/GOA/ENS/96M-02

REPLICATIVE USE OF AN EXTERNAL MODEL  
IN SIMULATION VARIANCE REDUCTION

THESIS

Thomas H. Irish  
Captain, USAF

AFIT/GOA/ENS/96M-02

Approved for public release; distribution unlimited

DTIC QUALITY INSPECTED 2

AFIT/GOA/ENS/96M-02

REPLICATIVE USE OF AN EXTERNAL MODEL  
IN SIMULATION VARIANCE REDUCTION

THESIS

Presented to the Faculty of the Graduate School of Engineering  
of the Air Force Institute of Technology  
Air University  
In Partial Fulfillment of the  
Requirements for the Degree of  
Master of Science in Operations Research

Thomas H. Irish, B.A.  
Captain, USAF

March, 1996

Approved for public release; distribution unlimited

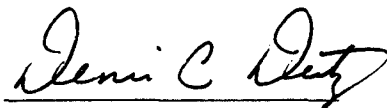

## THESIS APPROVAL

**NAME:** Thomas H. Irish, Captain, USAF

**CLASS:** GOA-96M

**THESIS TITLE:** Replicative Use of an External Model in Simulation Variance Reduction

**DEFENSE DATE:** 27 February 1996

COMMITTEE:	NAME/TITLE/DEPARTMENT	SIGNATURE
Advisor	Dennis C. Dietz, Lt Col, USAF Assistant Professor Department of Operational Sciences	
Reader	Kenneth W. Bauer Jr., Lt Col, USAF Associate Professor Department of Operational Sciences	

## *Acknowledgments*

I would like to thank all my classmates of GOA-96M for all their help and encouragement throughout the entire AFIT program. Particularly important was the humor they helped me find in the most trying of times.

I want to especially thank my thesis advisor, Lt Col Dennis C. Dietz. His guidance was invaluable. He allowed me to take ownership of the thesis process while providing me with just the right amount of help and just the right amount of push when I needed it. I would also like to thank him for suggesting this topic for research. I found the topic and the work involved very interesting (most of the time).

I also wish to thank my reader, Lt Col Kenneth W. Bauer Jr. His enjoyment of the topic was infectious. The committee meetings we had provided me with the direction I needed, and more importantly, the things I needed to think about.

Finally, I must thank my wife, Gloria, and my two boys, Kevin and Sean, for everything they've done. My family held me together throughout the entire process. Their love and support was always given without my asking. Their patience with me has been infinite. And of course, I must thank my Mom and Dad for the faith they maintained in me despite my efforts to destroy it.

## *Table of Contents*

	Page
Acknowledgments .....	iii
Table of Contents .....	iv
List of Figures .....	viii
List of Tables .....	ix
Abstract .....	xi
I. Introduction .....	1-1
1.1 Statement of the Problem .....	1-1
1.2 Background .....	1-3
1.2.1 Control Variates .....	1-3
1.2.2 Internal and External Control Variates .....	1-5
1.2.3 External Model (Analytical) Control Variates .....	1-6
1.3 Objective .....	1-7
1.3.1 Approach .....	1-8
1.3.2 Scope .....	1-9
II. Previous Work .....	2-1
2.1 Introduction .....	2-1
2.2 General Control Variate Methodology .....	2-1
2.2.1 Law and Kelton (1991) .....	2-1
2.2.2 Nelson (1987) .....	2-1

2.2.3	Lavenberg and Welch (1981) .....	2-2
2.3	External Control Variates .....	2-3
2.3.1	Burt, Gaver, and Perlas (1970) .....	2-3
2.3.2	Gaver and Shedler (1971) .....	2-4
2.4	Internal Control Variates .....	2-4
2.4.1	Lavenberg, Moeller, and Welch (1982) .....	2-4
2.4.2	Wilson and Pritsker (1984) .....	2-5
2.4.3	Bauer and Wilson (1993) .....	2-6
2.5	Analytical Control Variates .....	2-7
2.5.1	Tomick (1988) and Tomick, Litko, and Bauer (1990) .....	2-7
2.5.2	Dietz and Harmonosky (1989) .....	2-7
III.	Methodology .....	3-1
3.1	Introduction .....	3-1
3.2	Control Variate Method of Independent Replications .....	3-1
3.3	Open Queueing Network .....	3-4
3.3.1	Internal Control Variates (Open Queueing Network) .....	3-8
3.3.2	Analytical Control Variate (Open Queueing Network) .....	3-10
3.3.3	External Control Variate (Open Queueing Network) .....	3-12
3.4	Closed Queueing Network .....	3-14
3.4.1	Internal Control Variates (Closed Queueing Network) .....	3-20

3.4.2	Analytical Control Variate (Closed Queueing Network) .....	3-22
3.4.3	External Control Variate (Closed Queueing Network) .....	3-24
IV.	Results .....	4-1
4.1	Introduction .....	4-1
4.2	Experimental Procedures .....	4-1
4.2.1	Comparison Techniques .....	4-2
4.2.2	Open Queueing Network Settings .....	4-4
4.2.3	Closed Queueing Network Settings .....	4-5
4.3	Computer Implementation .....	4-7
4.4	Open Queueing Results .....	4-9
4.5	Closed Queueing Results .....	4-13
4.5.1	Confidence Interval Width Reduction .....	4-13
4.5.2	Estimation of Bias .....	4-20
V.	Conclusions and Recommendations for Future Research .....	5-1
5.1	Introduction .....	5-1
5.2	Study Efficiencies .....	5-1
5.3	Effectiveness over Different Model Configurations .....	5-4
5.4	System Insight .....	5-5
5.5	Loss Factor and Confidence Interval Width .....	5-6
5.6	Bias of Analytical Control Variate Point Estimates .....	5-7
5.7	Recommendations for Future Research .....	5-10



Appendix A. Computer Source Code .....	A-1
A.1 SLAM II Source Code for Model $M_1$ .....	A-1
A.2 FORTRAN Subroutine for Model $M_1$ .....	A-3
A.3 SLAM II Source Code for Model $M_3$ .....	A-5
A.4 FORTRAN Subroutine for Model $M_3$ .....	A-6
A.5 SLAM II Source Code for Models $Q_1$ and $Q_C$ .....	A-8
A.6 FORTRAN Subroutine for Models $Q_1$ and $Q_C$ .....	A-13
A.7 SLAM II Source Code for Model $Q_3$ .....	A-16
A.8 FORTRAN Subroutine for Model $Q_3$ .....	A-19
Appendix B. MATLAB Script Files .....	B-1
B.1 MATLAB Main Program for Analysis of Simulation Data .....	B-1
B.2 MATLAB Function for Calculating Controlled Responses .....	B-3
B.3 MATLAB Function for Computing Uncontrolled Response and Statistics .....	B-4
B.4 MATLAB Function to Average Statistics over all Experiments .....	B-5
B.5 MATLAB Function for Returning Appropriate Binary Matrix .....	B-6
Bibliography .....	Bib-1
Vita .....	Vita-1

## *List of Figures*

Figure	Page
3.1. Open queueing system model $M_1$ .....	3-5
3.2. Typical service time distributions and typical interarrival distribution .....	3-6
3.3. Open M/M/1 queueing network model $M_2$ and $M_3$ .....	3-11
3.4. Closed queueing network model $Q_1$ .....	3-14
3.5. Closed queueing network model $Q_C$ .....	3-17
4.1. $\bar{T}(20)$ experimental results with estimated confidence interval .....	4-28
4.2. $\bar{T}(\hat{\beta})^{ANALYT}$ experimental results with estimated confidence interval ...	4-29
4.3. $\bar{T}(\hat{\beta})^{EXT}$ experimental results with estimated confidence interval .....	4-29
4.4. $\bar{U}(20)$ experimental results with estimated confidence interval .....	4-30
4.5. $\bar{U}(\hat{\beta})^{ANALYT}$ experimental results with estimated confidence interval ..	4-31
4.6. $\bar{U}(\hat{\beta})^{INT}$ experimental results with estimated confidence interval .....	4-31
5.1. Histogram of 1,000 analytical control variate values .....	5-8

## *List of Tables*

Table	Page
3.1. All possible combinations of internal control variates .....	3-10
4.1. Experimental settings for open queueing network .....	4-4
4.2. Transition probability matrix values for closed queueing network .....	4-6
4.3. Service time distribution settings for the closed queueing network ...	4-6
4.4. Design points for closed queueing network .....	4-7
4.5. Confidence interval width reduction (%) for open queueing network .....	4-9
4.6. Realized coverage (%) for open queueing network .....	4-10
4.7. Estimated Mean Square Error for open queueing network .....	4-10
4.8. Performanc of 5 best internal control variate combinations for open queueing network .....	4-13
4.9. Confidence interval width reduction for system sojourn time for model $Q_1$ .....	4-15
4.10. Confidence interval width reduction for system sojourn time for model $Q_C$ .....	4-15
4.11. Confidence interval width reduction for CPU utilization for model $Q_1$ .....	4-16
4.12. Confidence interval width reduction for CPU utilization for model $Q_C$ .....	4-16
4.13. Confidence interval width reduction (%) for CPU utilization .....	4-18
4.14. Output from ForQue computer program .....	4-19
4.15. Realized coverage for system sojourn time for model $Q_1$ .....	4-21
4.16. Realized coverage for system sojourn time for model $Q_C$ .....	4-21

4.17. Realized coverage for CPU utilization for model $Q_1$ .....	4-22
4.18. Realized coverage for CPU utilization for model $Q_C$ .....	4-22
4.19. Estimated Mean Square Error for system sojourn time for model $Q_1$ .....	4-23
4.20. Estimated Mean Square Error for system sojourn time for model $Q_C$ .....	4-23
4.21. Estimated Mean Square Error ( $* 10^{-5}$ ) for CPU utilization for model $Q_1$ .....	4-24
4.22. Estimated Mean Square Error ( $* 10^{-5}$ ) for CPU utilization for model $Q_C$ .....	4-24
4.23. Estimated system sojourn time for 25,000 replications vs analytical controlled response for 20 replications .....	4-26
4.24. Estimated CPU utilization for 25,000 replications vs analytical controlled response for 20 replications .....	4-26
4.25. Comparison of estimated system sojourn time and mean value analysis .....	4-27
4.26. Comparison of estimated CPU utilization and mean value analysis ....	4-27

*Abstract*

The use of control variates is a well-known variance reduction technique for discrete event simulation experiments. Currently, internal control variates are used almost exclusively by practitioners and researchers when using control variates. The primary objective of this study is to explore the variance reduction achieved by the replicative use of an external, analytical model to generate control variates. Performance for the analytical control variates is compared to the performance of typical internal and external control variates for both an open and a closed queueing network. Performance measures used are confidence interval width reduction, realized coverage, and estimated Mean Square Error. Results of this study indicate analytical control variates achieve comparable confidence interval width reduction with internal and external control variates. However, the analytical control variates exhibit greater levels of estimated bias. Possible causes and remedies for the observed bias are discussed and areas for future research and use of analytical control variates conclude the study.

# REPLICATIVE USE OF AN EXTERNAL MODEL IN SIMULATION VARIANCE REDUCTION

## *1. Introduction*

### *1.1 Statement of the Problem*

Air Force and industry analysts often use computer simulations to study large scale systems. Decision makers use the results of these simulations to set policy and allocate scarce resources. In many applications, analysts focus on one measurement of effectiveness as the basis for evaluating competing policies or allocations. Obviously, the better an analyst can estimate this measure, the better the final decision.

Simulations produce random outputs, not exact answers. To interpret a simulation output, analysts typically use statistical confidence intervals based on the mean and variance of the output random variable in order to estimate its true expected value. Analysts may not have the time or budget to complete enough replications to produce a sufficiently small confidence interval. By using an appropriate variance reduction technique, the analyst can significantly shrink the size of the confidence interval or reduce the computational effort needed to obtain an estimate of specified accuracy.

Control variates comprise one method that has been shown to significantly reduce variance. Control variates are usually classified as either internal or external control variates. Both methods attempt to take advantage of correlation between selected random variables to reduce variance. Internal control variates can be the input random variables, or simple functions of them. Unfortunately, selecting a subset of all the possible internal

control variates for a given simulation study that produces the small and reliable confidence intervals can be a difficult problem [14]. External control variates require the creation of a simplified analytical model of the system under study and an associated simulation model based on the analytical model. Applying external control variates to large simulations can be quite difficult since common random numbers must be applied to each simulation [13]. For these reasons, external control variates are rarely used in practical applications [17].

By using an external, analytical model to generate a single control variate for each replication of a simulation, it may be possible to avoid both the undesirable impact of using multiple control variates and the difficult application of common random numbers. Sharon [19] first used Jackson networks to generate control variates for queueing network simulations. Tomick, Litko, and Bauer [21] demonstrated that in small queueing networks significant variance reduction can be achieved by generating control variates using an analytical model. Dietz and Harmonosky [6] achieved significant variance reduction in an aircraft sortie generation simulation by using a composite control variate representing a total expected "hands-on" time per sortie. If this same technique can reduce variance in large real world simulations, study times can be significantly reduced. This study will explore the variance reduction achieved with the replicative use of an external model to generate control variates when applied to simulations of varying sizes. The variance reduction achieved using this method will be compared to the variance reduction obtained in the same simulations using typical internal and external control variate techniques.

## 1.2 Background

The following background information is provided as motivation for the development of this study. First, the theory of control variates (particularly how they reduce variance and how simulation confidence intervals are formed) is presented, followed by a discussion of internal and external control variates. The theoretical basis for using an external model to generate a single control variate is discussed next.

*1.2.1 Control Variates.* This variance reduction technique attempts to take advantage of the correlation between different random variables in order to obtain a tighter estimate of a response variable. For example, suppose an analyst is trying to estimate  $\mu = E[Y]$ , where the random variable  $Y$  is the waiting time for a customer in a bank line. To reduce the variance, the analyst can then use another random variable, say  $C$ , that the analyst believes to be correlated to  $Y$  and that has a known expectation  $\mu_C = E[C]$ . A new random variable  $Y(b) = Y - b(C - \mu_C)$  can be constructed for each replication where

$$E[Y(b)] = E[Y] - b(E[C] - \mu_C) \quad (1.1)$$

is an unbiased estimator for  $\mu$  for any real number  $b$ . Then since,

$$Var(Y(b)) = Var(Y) + b^2 Var(C) - 2b Cov(Y, C) \quad (1.2)$$

$Var(Y(b))$  will have a smaller value than  $Var(Y)$  if and only if

$$2a Cov(Y, C) > b^2 Var(C) \quad (1.3)$$



which will be the case if  $Y$  and  $C$  are highly correlated and  $b$  is selected appropriately. Let  $\beta$  be the optimal value for  $b$  that minimizes  $Var(Y(b))$ . By setting the derivative of equation (1.2), (with respect to  $b$ ), equal to zero we find

$$\beta = \frac{Cov(Y, C)}{Var(Y)} \quad (1.4)$$

Normally  $Cov(Y, C)$  and  $Var(Y)$  are not known. Then for  $n$  replications,  $\beta$  can be estimated by

$$\hat{\beta} = \frac{\sum_{j=1}^n (Y_j - \bar{Y})(C_j - \bar{C})}{\sum_{j=1}^n (Y_j - \bar{Y})^2} \quad (1.5)$$

where  $\bar{Y}$  is the mean of the  $n$  observations of  $Y$  and  $\bar{C}$  is the mean of the  $n$  observations of  $C$ . Then for the same  $n$  replications we can estimate  $\mu$  with the controlled estimate

$$\bar{Y}(\hat{\beta}) = \bar{Y} - \hat{\beta}(\bar{C} - \mu_c) \quad (1.6)$$

This method can be generalized for several control variates,  $C_1, C_2, \dots, C_q$  with respective means  $\mu_1, \mu_2, \dots, \mu_q$  to

$$Y(b) = Y - \sum_{k=1}^q b_k (C_k - \mu_k) \quad (1.7)$$

where the  $b_k$ 's are any real numbers. If the  $b_k$ 's are estimated in the same manner as equation (1.6), the variance of  $\bar{Y}(\hat{\beta})$  is given by [14]

$$Var[\bar{Y}(\hat{\beta})] = \frac{n-2}{n-q-2} (1 - R^2) Var[\bar{Y}] \quad (1.8)$$

and the associated confidence interval is

$$\bar{Y}(\hat{\beta}) \pm t_{1-\alpha/2, n-q-1} \sqrt{\frac{(n-2)}{(n-q-2)} (1-R^2) \text{Var}[\bar{Y}]} \quad (1.9)$$

where  $R^2$  is the coefficient of correlation between  $Y$  and  $C = [C_1, \dots, C_q]$ ,  $n$  is the number of replications, and  $q$  is the number of control variates. (Note equations (1.8) and (1.9) are not computational formulas. The computational formulas are given in chapter 3.) As each additional control variate is added, the  $t$ -statistic loses an additional degree of freedom and the loss term  $(n-2)/(n-q-2)$  increases, while  $R^2$  may increase. Due to these countering effects, at some point the addition of another control variate can cause the confidence interval to grow rather than decrease. [13]

*1.2.2 Internal and External Control Variates.* Control variates can be obtained either internally or externally. Internal control variates arise from the simulation itself and are normally some combination of the input random variables or simple functions of them. Examples are interarrival times, service times, and demand levels since their expectations are known and analysis of the system should provide the sign of the correlation between them and the output random variable. As demonstrated in the previous section, as more control variates are added to induce more correlation, the corresponding confidence interval will begin to grow at some point. Selection of the best subset of all possible internal control variates then becomes a problem of its own [12].

External control variates are normally obtained by simultaneously simulating a similar system that has an analytical solution. By using common random numbers the output variable of the simple simulation should be positively correlated to the output variable of

the system under study (hopefully highly correlated). Using external control variates is usually not “cheap” due to the extra work required to develop a similar, simpler system, the effort required to synchronize the common random numbers, and the additional computer time required to run the external simulation [6]. However, it can pay off if the resulting variance reduction is larger than that of the internal control variate.

*1.2.3 External Model (Analytical) Control Variates.* Since finding the “best” subset of internal control variates or synchronizing common random can be difficult tasks in large simulations, control variates are often not considered. However, Tomick et al. [21] and Dietz and Harmonosky [6] both demonstrated that control variates could be obtained directly from an analytical model. The control variates obtained in each paper can be viewed simply as new random variables that are functions of the input random variables. This type of control variate can be considered a hybrid of both internal and external control variates since it is found using an external model -- like external control variates -- and is a function of the input random variables that doesn't require another simulation model -- like internal control variates. In order to avoid confusion, control variates found in this manner will be referred to as analytical control variates throughout the rest of this study.

To illustrate how an analytical control variate can be generated consider a simulation model where  $Y$  is the output random variable of interest with  $E[Y] = \mu$ . Also, let  $\mathbf{X} = (X_1, X_2, \dots, X_m)$  be the vector of input random variables that drive the simulation model with  $E[\mathbf{X}] = (\mu_1, \mu_2, \dots, \mu_m)$ . An appropriate analytical model is essentially a function of some subset of the same input random variables that drive the simulation, ideally all of the input

random variables. This function should then provide an approximation of the output of the simulation. To illustrate, represent the analytical model as the function  $f(X) = Z$ . Let  $E[f(X)] = f(\mu_1, \dots, \mu_m) = Z_\mu$ . Then a new random variable, the analytical control variate, can be constructed as before by  $Y(b) = Y - b(f(X) - Z_\mu)$ , so that

$$E[Y(b)] = E[Y] - b(E[f(X)] - Z_\mu) \quad (1.10)$$

is an unbiased estimator of  $\mu$  for any real number  $b$ .

To generate analytical control variate point estimates let  $\bar{X}_j$  be the mean of the observed values of  $X$  for the  $j$ -th replication. Further let  $f(\bar{X}_j) = Z_j$  be the value of the analytical model for the  $j$ -th replication and let  $\bar{Z}$  be the mean of these calculations for all  $n$  replications performed. Then an analytically controlled estimate of  $\mu$  is

$$\bar{Y}(\hat{\beta}) = \bar{Y} - \hat{\beta}(\bar{Z} - Z_\mu) \quad (1.11)$$

where  $\hat{\beta}$  is estimated using equation (1.5). As long as  $f(\bar{X}_j)$  and  $Y_j$  are correlated variance reduction should occur.

### 1.3 Objective

The primary objective of this study is to investigate the variance reduction of the replicative use an external model to generate analytical control variates for simulations. This study will focus on simulations with a single random variable output. The primary performance characteristic for this study will be confidence interval width rather than variance. Confidence interval width is chosen for two reasons. First, analysts are primarily concerned with confidence interval width when performing simulation studies.

Secondly, confidence interval width is a function of variance, the number of control variates chosen, and number of replications. Thus, internal control variates can be compared to external and analytical control variates in a meaningful manner. Comparisons of the point estimate confidence interval width will be made when no control variates are used to the point estimate confidence interval width when internal control variates, external control variates, and analytical control variates are used on the same simulation. In addition, realized confidence interval coverage and estimated mean square error will be compared for all cases to investigate the possibility of bias error.

*1.3.1 Approach.* The objectives of this study will be accomplished in the following manner.

1. Create a simple open queueing network and an appropriate simulation model.  
Develop an external analytical model whose output is correlated to the output of the simulation model. Create a separate simulation model of the analytical model.
2. Perform simulation experiments on three different design points of the open queueing network and generate internal, analytical, and external control variates. Calculate response statistics without controls and with controls and compare the resulting confidence intervals.
3. Create or find a more complex closed queueing network and build an appropriate simulation model. Using the mean value analysis algorithm as the analytical model, create a simulation program of the analytical model.
4. Perform simulation experiments on twelve different design points of the closed queueing network and generate internal, analytical, and external control variates.

Calculate response statistics without controls and with controls and compare the resulting confidence intervals.

*1.3.2 Scope.* Only estimates of single random variable confidence intervals will be explored. Estimating the reduction in multivariate confidence intervals will not be discussed. Estimating simulation outputs can be accomplished using replication methods, batch means, or the regenerative method. Only the replication method will be considered in this study. Other variance reduction techniques such as common random numbers, antithetic variates, indirect estimation, or conditional expectations will not be considered.

## ***2. Previous Work***

### ***2.1 Introduction***

A review of the current literature on the use of control variates to reduce the variance of simulation studies as it pertains to this study is presented below. The discussion will focus on the use of control variates when estimating single variate responses using replicative simulation methods. The first section presents some of the basic references on control variates, followed by a review on the use of external control variates. Selected topics on the use of specific internal control variates as applicable to this study are presented next. The final section reviews the literature on analytical control variates

### ***2.2 General Control Variate Methodology***

***2.2.1 Law and Kelton (1991).*** Several authors present excellent references on the use of control variates to reduce variance. The Law and Kelton textbook (1991) on simulation modeling includes a section on the use of control variates in their chapter of variance reduction techniques. They provide an overview on control variate theory, application, and a brief discussion on the difference between internal and external control variates. In addition to their excellent coverage of control variates, they also provide an extensive reference list of control variate literature.

***2.2.2 Nelson (1987).*** This article presents a guide for simulation practitioners for applying three variance reduction techniques, including control variates. Nelson provides methods for finding point and interval estimators, software requirements, and guidelines

for experiment design when using control variates. This article is designed as a tutorial, not as a definitive presentation of control variate theory. As such it is very useful in understanding how the theory of control variates is applied to actual simulation studies. For example, Nelson points out that the computation of control variate point and interval estimators is the same as estimating the intercept term of a least-squares regression of the responses on the control variates minus their known means.

Nelson goes on to describe two methods for deciding which subset of possible control variates to use for reducing variance. (It should be noted that when Nelson refers to control variates, he is referring to only internal control variates.) One method is to use a regression software package and perform stepwise regression on all possible control variates and then select the subset of controls that create the largest amount of variance reduction. He also proposes a means of determining a marginal improvement ratio for adding an additional control variate to the set of controls already in use. For example, for a set of  $q$  control variates, the marginal improvement ratio is computed by comparing  $1 - R^2$  for the set of  $q + 1$  control variates, and  $1 - R^2$  for  $q$  control variates, where  $R^2$  is an estimate of the square of the multiple correlation coefficient of the response variable. He then provides a table of marginal improvement ratios necessary for adding an additional control variate based on the number of replications performed and number of control variates already included.

*2.2.3 Lavenberg and Welch (1981).* This is a very detailed survey on the application of control variates. The methods they present in generating confidence intervals form the foundation of the methods used in this study. As in Nelson (1987), the focus is on internal



control variates, but Lavenberg and Welch do point out that control variates can be obtained from the simulation of a second related model that has a known expected value and is driven by the same sequences of random numbers that are used in the simulation under study. These control variates can then be looked upon as complex functions of the input random variables and the theory of control variates they present holds for these types of control variates too.

Other areas investigated include techniques for generating control variates (internal), inefficiencies resulting from estimating control variate coefficients, a review of the studies completed at the time of the article, and directions for future research. Particularly useful are the two appendices of this article which provide detailed discussions on the application of control variates to generate confidence intervals. The first appendix describes the generation of confidence intervals using the method of independent replications, (and equivalently the method of batch means) while the second appendix presents the generation of confidence intervals using the regenerative method of simulation.

### *2.3 External Control Variates*

*2.3.1 Burt, Gaver, and Perlas (1970).* The effectiveness of several variance reduction techniques on project graph analysis (PERT, GERT, CPM, etc.) network simulations is examined in this article. The techniques considered are antithetic variates, stratification, and what we would now call external control variates. They generate the external control variates by first constructing simplified networks that have analytical solutions and are "similar" to the networks under study. They then simulate both models

separately using the same random number stream to drive both simulations. The generated control variates are then applied in two ways. First, no effort is made to determine an optimal control variate coefficient, and therefore "-1" is used to calculate the controlled response. They call this straight control estimates. Burt et.al. then use the regression technique to estimate the optimal coefficient. Two different networks are simulated with control variates and results are reported of significant variance reduction with the regression method being the most effective.

2.3.2 *Gaver and Shedler (1971)*. The application of external control variates to simulations of a multiprogrammed computer system is examined by the authors. In a similar manner to the study above, Gaver and Shedler propose a model of a multiprogrammed computer system and a similar model that can be solved analytically. In the same manner as above, both models are simulated and the generated control variates are applied using both the straight control variate and regression to estimate an optimum coefficient. Results indicate significant reduction in variance with the best results from the use of regression.

## 2.4 *Internal Control Variates*

2.4.1. *Lavenberg, Moeller, and Welch (1982)*. The authors propose three different internal control variates (service time variable, flow variable, and work variable) that can be used in simulation studies of a wide class of closed queueing networks. The closed networks under study consist of a finite number of service centers with one or multiple servers. Additionally, there is a fixed number of customers of different classes. Service

time variables are the sample mean service time for a particular class of customers at a specific service center. The fraction of service completions at a specific service center that are for a particular class of customers is defined as a flow variable. The product of these two variables, the sum of service times for a class of customers at a service center, constitute the work variables. Lavenberg et. al. then perform experiments on a simple model of an interactive multiprogrammed computer system. This same model is used in this study to assess the performance of analytical control variates on closed queueing networks. Results from their experiments show work variables produce significant reduction in variance over several different experimental configurations.

2.4.2 *Wilson and Pritsker (1984)*. Standardized work variables are developed for queueing systems with the regenerative property in this study. Given a service process  $\{U_j(k): j \geq 1\}$ , at service center  $k$  with known expected value  $\mu_k$  and variance of  $\sigma_k^2$ , standardized work variables are defined as

$$C_k(t) = [a(k,t)]^{-1/2} \sum_{j=1}^{a(k,t)} [U_j(k) - \mu_k] / \sigma_k$$

where  $a(k,t)$  is the number of service times started at center  $k$  during the time period  $[0,t]$ . These standardized work variables are used later in this study and are defined for use with the models under study in sections 3.3.1 and 3.4.1 below. A need for these standardized variables was seen since the work variables presented in Lavenberg (1982) have been shown to have an asymptotic variance equal to zero which causes the variance covariance matrix for a set of these controls to be asymptotically singular. Since an inverse of this matrix (or usually an estimate of it) must be computed to find (or estimate) the optimal

control coefficient, numerical problems can arise as replication length increases. The standardized work variables, on the other hand, have an asymptotic variance of 1. Experiments on a simple network were performed with substantial variance reduction reported.

2.4.3 *Bauer and Wilson (1993)*. Another set of standardized control variables, standardized routing variables, are developed which can be used for discrete-event simulation models that use a multinomial construct. These control variates attempt to exploit the correlation between departures from the mean branching probabilities in a network and the resultant network response. Standardized routing variables are defined in the following manner. Consider a multinomial branching process of  $g$  branches leading to  $g$  service centers, and define an indicator variable as

$$I_i(j) = \begin{cases} 1 & \text{if the } i\text{-th departing customer goes to center } j, \\ 0 & \text{otherwise.} \end{cases}$$

Then a standardized routing variable for center  $j$  is defined as

$$R_j = \sum_{i=1}^{N(t)} \frac{I_i(j) - p(j)}{\{N(t)[1 - p(j)]p(j)\}^{1/2}}, \quad j = 1, \dots, g.$$

where  $N(t)$  is the total number of transits through all  $g$  branches in the time interval  $[0, t]$  and  $p(j)$  is the probability that a customer is sent to center  $j$ . Standardized routing variables are used in this study and are defined in terms of the study model in section 3.4.1 below. Bauer and Wilson performed several experiments on a simple network with results indicating significant confidence interval length reduction, particularly when the standardized routing variables are used in conjunction with standardized work variables.

## 2.5 Analytical Control Variates

2.5.1 *Tomick (1988) and Tomick, Litko, and Bauer (1989)*. Both of these works report variance reduction achieved for external control variates that are obtained by solving approximate queueing models. These types of control variates are called analytical in this study to differentiate them from external control variates obtained via the simulation of a separate simulation model. Throughout the rest of this discussion, Tomick's external control variates will be called analytical control variates to maintain consistency in this study. Tomick computes the analytical control variates two ways. The first method consists of solving a related Jackson network using the observed values of the input random variables for each replication. The second method uses the Queueing Network Analyzer software developed by Whitt [22] with inputs of the observed values of the random variables for each replication. Experiments were conducted on a simple open queueing network in Tomick (1988) that indicated analytical control variates achieved significant levels of variance reduction, however confidence interval coverages were less than satisfactory indicating a bias problem with the analytical control variates under study.

2.5.2 *Dietz and Harmonosky (1989)*. This paper examines the applicability of a single internal control variate that is formulated by aggregating many of the key input random variables for use with a simulation model of an aircraft sortie generation model. The control variate used is an analytical model of the "hands-on" time between each sortie, which includes all ground activities and flying time with inputs being the observed parameters from the simulation run. Obviously this single internal control variate is the same as the analytical control variate referred to in this study. Several experiments were

run for different values of aircraft reliability, aircraft maintainability and maintenance resource availability. Results indicate significant variance and confidence interval length reduction can be obtained using this approach. Results on confidence interval coverage are not reported.

### ***3. Methodology***

#### ***3.1 Introduction***

The performance of analytical control variates will be compared to that of various internal and external control variates. This will be accomplished on several configurations of two queuing network models. All control variates will be compared using the Method of Independent Replications [12] as described in section 3.2. The method is commonly referred to as the "regression method" since linear regression formulas can be used to find controlled responses as well as estimate controlled variance values and their associated confidence intervals.

The following sections present the two queueing networks and the particular statistical performance measurements for each model. The first section addresses a simple open queueing network and the second deals with a more complex closed queueing network. Each section includes model-specific discussion on the selection of internal control variates, external models for external control variates, and analytical control variates.

#### ***3.2 Control Variate Method of Independent Replications.***

The following methods will be used to calculate controlled responses and estimate controlled response variances and confidence intervals. First define  $\mu$  as the expected value of a system response measurement which we wish to estimate and  $Y_j$  as an unbiased

estimator of  $\mu$  for the  $j$ -th replication of a simulation. Then define

$$\hat{\mu} = \bar{Y}(n) = \frac{1}{n} \sum_{j=1}^n Y_j \quad (3.1)$$

Let  $C = (C_1, C_2, \dots, C_q)'$  be a  $q \times 1$  vector of control variates with the known mean vector of  $\mu_C = (\mu_1, \mu_2, \dots, \mu_q)'$ . Let  $B = (b_1, b_2, \dots, b_q)'$  be a  $q \times 1$  vector of constants.

Then a controlled estimate of an expected response,  $\mu$ , is given by [13]

$$\bar{Y}(B) = \bar{Y}(n) - B'(\bar{C} - \mu_c) \quad (3.2)$$

The variance of  $\bar{Y}(B)$  is minimized by the optimal vector of control coefficients given by

$$\beta = \sigma_{YC} \Sigma_{CC}^{-1} \quad (3.3)$$

where  $\sigma_{YC}$  is the covariance vector between  $Y$  and  $C$ , and  $\Sigma_{CC}$  is the covariance matrix of  $C$ . Since these values are normally unknown,  $\beta$  is estimated by

$$\hat{\beta} = S_{YC} S_{CC}^{-1} \quad (3.4)$$

where

$$\begin{aligned} S_{YC} &= \frac{1}{(n-1)} \sum_{j=1}^n (Y_j - \bar{Y})(C_j - \bar{C}) \\ S_{CC} &= \frac{1}{(n-1)} \sum_{j=1}^n (C_j - \bar{C})(C_j - \bar{C})' \end{aligned} \quad (3.5)$$

Then a controlled estimate of  $\mu$  is given by

$$\bar{Y}(\hat{\beta}) = \bar{Y}(n) - \hat{\beta}'(\bar{C} - \mu_c) \quad (3.6)$$

If we assume both  $Y$  and  $C$  have the following joint normal distribution

$$\begin{bmatrix} Y \\ C \end{bmatrix} \sim N_{1+q} \left[ \begin{bmatrix} \mu_Y \\ \mu_C \end{bmatrix}, \begin{bmatrix} \sigma_Y^2 & \sigma_{YC} \\ \sigma_{CY} & \Sigma_{CC} \end{bmatrix} \right] \quad (3.7)$$



then  $\bar{Y}(\hat{\beta})$  is an unbiased estimator of  $\mu$  and the estimated variance is given by [2]

$$S^2(\bar{Y}(\hat{\beta})) = D^2 S_{Y.C}^2 \quad (3.8)$$

where

$$D^2 = \frac{1}{n} + \frac{1}{(n-1)} [(\bar{C} - \mu_c)' S_{CC}^{-1} (\bar{C} - \mu_c)] \quad (3.9)$$

and

$$S_{Y.C}^2 = \frac{n-1}{n-q-1} [\hat{\sigma}_Y^2 - S_{YC} S_{CC}^{-1} S_{YC}'] \quad (3.10)$$

Then a  $100(1-\alpha)\%$  confidence interval is given by

$$\bar{Y}(\hat{\beta}) \pm t_{1-\alpha/2, n-q-1} D S_{Y.C} \quad (3.11)$$

Since we have assumed  $Y$  and  $C$  to have a multivariate normal distribution, the application of control variates using  $\hat{\beta}$  can be considered a classical regression problem [9]. Computationally, this can be a simpler method for finding  $\bar{Y}(\hat{\beta})$  and estimating its variance and associated confidence interval. Since

$$E[Y|C=c] = \mu + \beta'(c - \mu_c) \quad (3.12)$$

the problem can be stated as

$$\begin{aligned} Y &= X\gamma + \varepsilon \\ \text{where } Y' &= [Y_1, Y_2, \dots, Y_r] \\ \text{and } \gamma &= \begin{bmatrix} \mu \\ \beta \end{bmatrix} \end{aligned} \quad (3.13)$$

with

$$X = \begin{bmatrix} 1 & c_{11} - \mu_1 & \dots & c_{q1} - \mu_q \\ \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots \\ 1 & c_{1n} - \mu_1 & \dots & c_{qn} - \mu_q \end{bmatrix} \quad (3.14)$$

Then to find the least squares estimators set

$$\begin{bmatrix} \hat{\mu} \\ \hat{\beta} \end{bmatrix} = (X'X)^{-1} X'Y \quad (3.15)$$

Since [11]

$$\bar{Y}(n) = \hat{\mu} + \hat{\beta}'(\bar{C} - \mu_c) \quad (3.16)$$

it follows that  $\hat{\mu}$  is equal to  $\bar{Y}(\hat{\beta})$ . Then from regression theory results for  $\hat{\mu}$  [16], the

estimated variance of  $\bar{Y}(\hat{\beta})$  is given by

$$Var(\bar{Y}(\hat{\beta})) = s_{11}MSE \quad (3.17)$$

where  $s_{11}$  is the upper left element of  $(X'X)^{-1}$  and MSE can be computed as

$$MSE = \left( \frac{1}{n - q - 1} \right) Y'Y - \gamma X'Y \quad (3.18)$$

Then the  $100(1-\alpha)\%$  confidence interval can be given by

$$\bar{Y}(\hat{\beta}) \pm t_{1-\alpha/2, n-q-1} \sqrt{s_{11}MSE} \quad (3.19)$$

### 3.3 Open Queuing Network.

The open queuing system studied consists of three server stations connected in series, (see figure 3.1). Call this model  $M_1$ . Each station, or node, has a dedicated single

server that can provide service to only one entity at a time. All arrivals enter the system at node 1 with balking not considered. Upon completion of service at node 1 all entities proceed on to nodes 2 and 3 consecutively. Entities depart the system after receiving service at node 3.

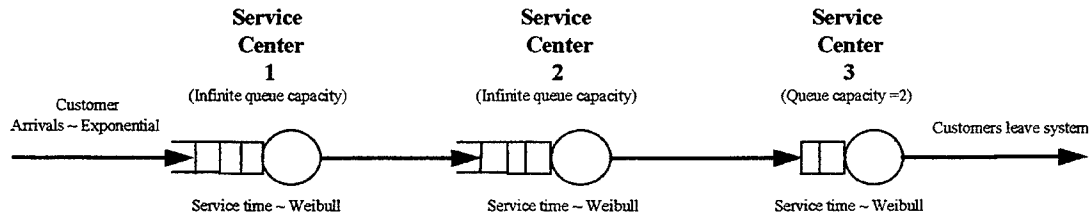


Figure 3.1. Open queueing system model  $M_1$

System interarrival times are IID exponential random variables with a mean interarrival time equal to  $\lambda$  (Poisson arrival process) and probability density function

$$f(x) = \begin{cases} \frac{1}{\lambda} e^{-x/\lambda} & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.20)$$

Each service node has a different service time distribution, but each is an IID Weibull random variable with probability density function

$$f(x) = \begin{cases} \frac{\alpha}{\beta} x^{\alpha-1} e^{-x^\alpha/\beta} & \text{if } x \geq 0; \alpha, \beta > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.21)$$

Node 1's service time distribution has shape parameter  $\alpha_1$  and scale parameter  $\beta_1$ , node 2's parameters are  $\alpha_2$  and  $\beta_2$ , and node 3's parameters are  $\alpha_3$  and  $\beta_3$ . Then the mean service time for each node is equal to

$$\mu_k = \beta_k^{1/\alpha_k} \Gamma\left(\frac{\alpha_k + 1}{\alpha_k}\right) \quad k = 1, 2, 3 \quad (3.22)$$

with a variance of

$$\sigma_k^2 = \beta_k^{2/\alpha_k} \left( \Gamma\left(\frac{\alpha_k + 2}{\alpha_k}\right) - \Gamma\left(\frac{\alpha_k + 1}{\alpha_k}\right)^2 \right) \quad k = 1, 2, 3 \quad (3.23)$$

(Typical service time and arrival probability density functions are shown in figure 3.2.)

Nodes 1 and 2 have an infinite queue capacity while node 3 has a finite queuing capacity of 2. If node 3's queue is full and an entity has finished service at node 2, that entity will wait at node 2 and block node 2's server until a spot becomes available in node 3's queue.

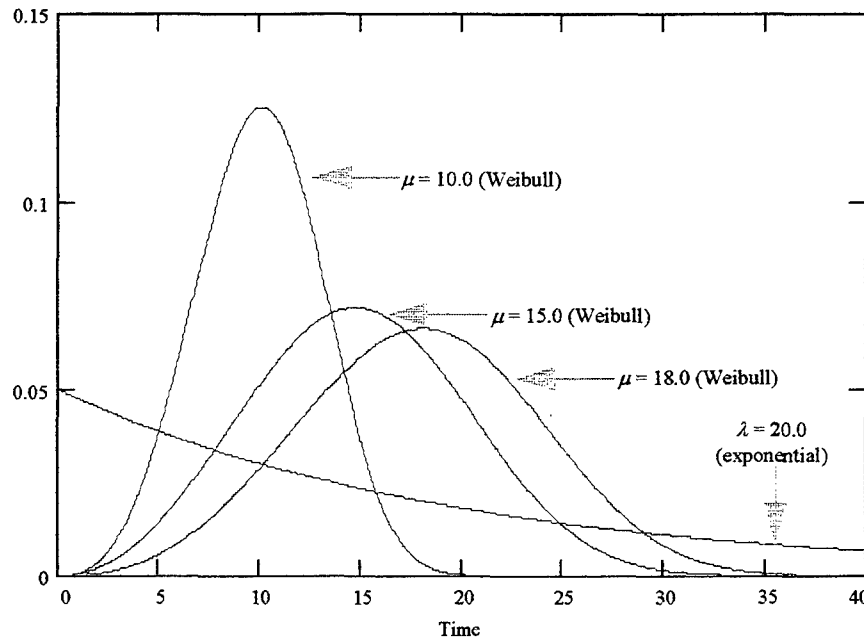


Figure 3.2. Typical service time distributions (Weibull:  $\mu = 10.0, 15.0, \text{ and } 18.0$ ) and typical interarrival distribution (exponential:  $\lambda = 20.0$ ).

This network is a simple model of a hypothetical fast-food restaurant drive through window system. The first node represents the order microphone, the second node

represents the pay window, and the third node represents the pick-up window. Entities being served within the system are fast-food drive through customers. The model is not meant to represent any specific facility, but has been created for illustrative purposes and the system assumptions have been made for ease of analysis.

The system performance measurement that will be estimated is the expected time a customer spends in the system (sojourn time) under steady state conditions. Let  $\mu$  be the true steady state expected sojourn time which we wish to estimate and let  $y_{ij}$  be the sojourn time of the  $i$ -th customer during the  $j$ -th replication. Also let  $l$  be the number of customers in the warm-up period,  $m$  be the total number of customers for each replication, and  $n$  the number of replications. Then, let

$$Y_j = \frac{1}{m-l} \sum_{i=l+1}^m y_{ij} \quad j = 1, 2, \dots, n \quad (3.24)$$

so that  $\mu$  can be estimated using the replication/deletion method [13] by

$$\hat{\mu} = \bar{Y}(n) = \frac{1}{n} \sum_{j=1}^n Y_j \quad (3.25)$$

which has an estimated variance of

$$\hat{\sigma}_{\bar{Y}(n)}^2 = \frac{S^2(n)}{n} = \frac{1}{n} \frac{\sum_{j=1}^n (Y_j - \bar{Y}(n))^2}{n-1} \quad (3.26)$$

and an approximate  $100(1-\alpha)$  percent confidence interval given by

$$\bar{Y}(n) \pm t_{n-1, 1-\alpha/2} \sqrt{\frac{S^2(n)}{n}} \quad (3.27)$$

3.3.1 *Internal Control Variates (Open Queueing Network)*. Internal control variates are random variables generated within the simulation that have known means. For the open queueing model, candidate random variables can be derived from the interarrival distribution and the three service time distributions. Standardized work variables, which are a function of the service time distribution at a specific service center, have been shown to be good control variate candidates [23]. Let the service time process at service center  $k$  be the IID sequence  $U_i(k): i \geq 1, k = 1, 2, 3$ ; (1 = order microphone, 2 = pay window, 3 = pickup window). Then a standardized work variable is defined by

$$W_{kj} = \frac{1}{\sqrt{m-l}} \sum_{i=l+1}^m \frac{U_{ij}(k) - \mu_k}{\sigma_k} \quad j = 1, 2, \dots, n \quad (3.28)$$

where  $\mu_k$  and  $\sigma_k$  are given by equations (3.22) and (3.23) respectively for model  $M_1$ . Each  $W_{kj}$  has a mean of zero as  $(m-l) \rightarrow \infty$  [23].

Another candidate internal control variate is a standardized interarrival variable. Similarly let  $V_i: i \geq 1$ , be the IID sequence of system interarrival times. Since  $\lambda$  is the mean of the interarrival time distribution for model  $M_1$ , define a standardized interarrival variable as

$$A_j = \frac{1}{\sqrt{m-l}} \sum_{i=l+1}^m \frac{V_{ij} - \lambda}{\sigma_a} \quad j = 1, 2, \dots, n \quad (3.29)$$

which also has a mean of zero as  $(m-l) \rightarrow \infty$  [23].

Given the four candidate internal control variates defined above, fifteen ( $2^4 - 1$ ) different combinations of control variates could be used to reduce the variance of the

estimated average sojourn time. Let  $C^{INT}$  be the matrix of all candidate internal control variates such that

$$C^{INT} = \begin{bmatrix} W_{11} & \dots & W_{1n} \\ W_{21} & \dots & W_{2n} \\ W_{31} & \dots & W_{3n} \\ A_1 & \dots & A_n \end{bmatrix} \quad (3.30)$$

Also let  $\bar{W}_k = \frac{1}{n} \sum_{j=1}^n W_{kj}$  and  $\bar{A} = \frac{1}{n} \sum_{j=1}^n A_j$ . Then let

$$\bar{C}^{INT} = \begin{bmatrix} \bar{W}_1 \\ \bar{W}_2 \\ \bar{W}_3 \\ \bar{A} \end{bmatrix} \quad (3.31)$$

be the vector of internal control variate means. Then for  $p = 1, 2, \dots, 15$ ,  $C_p^{INT}$  is one of the possible matrices of selected control variates and  $\bar{C}_p^{INT}$  is the associated vector of sample means. Table 1.1 enumerates the different internal control variate combinations. Let  $\bar{Y}(\hat{\beta})_p^{INT}$  be the internally controlled, estimated expected sojourn time using  $C_p^{INT}$ ,  $p = 1, 2, \dots, 15$ , so that

$$\bar{Y}(\hat{\beta})_p^{INT} = \bar{Y}(n) - \hat{\beta}'(\bar{C}_p^{INT} - \mathbf{0}) \quad (3.32)$$

Then each  $\bar{Y}(\hat{\beta})_p^{INT}$  can be calculated using equation (3.15) and the estimated variance and confidence intervals can be found using equations (3.17) and (3.19) respectively. Performance of each controlled estimator can be compared based on variance reduction and confidence interval reduction for each estimator.

$p$	$C_p^{INT}$
1	$[W_1]$
2	$[W_2]$
3	$[W_3]$
4	$[A]$
5	$[W_1 W_2]'$
6	$[W_1 W_3]'$
7	$[W_1 A]'$
8	$[W_2 W_3]'$
9	$[W_2 A]'$
10	$[W_3 A]'$
11	$[W_1 W_2 W_3]'$
12	$[W_1 W_2 A]'$
13	$[W_1 W_3 A]'$
14	$[W_2 W_3 A]'$
15	$[W_1 W_2 W_3 A]'$

Table 3.1. All possible combinations of internal control variates

3.3.2 *Analytical Control Variate (Open Queueing Network).* Application of analytical control variates requires the development of a separate model, say  $M_2$ , that can be solved analytically. Additionally, the solution to the analytical model  $M_2$  must be correlated to the response of each replication of the simulation model  $M_1$  in order to reduce the variance of the response using control variate techniques. In this specific case, the analytical model is a series M/M/1 open queueing network with three nodes as depicted in figure 3.3. The interarrival rate is  $1/\lambda$ , same as  $M_1$ , and the three service rates are given by  $1/\mu_1$ ,  $1/\mu_2$ , and  $1/\mu_3$ , with  $\mu_k$  given by equation (3.21). Based on parameters  $\alpha_k$  and  $\beta_k$



from model  $M_1$ . Then, from queuing theory, the expected steady state sojourn time is known to be [18]

$$\mu^{ANALYT} = \frac{1}{\frac{1}{\mu_1} - \frac{1}{\lambda}} + \frac{1}{\frac{1}{\mu_2} - \frac{1}{\lambda}} + \frac{1}{\frac{1}{\mu_3} - \frac{1}{\lambda}} \quad (3.33)$$

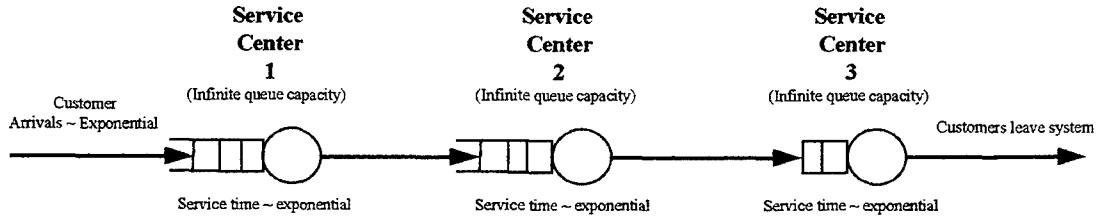


Figure 3.3. Open M/M/1 queueing network model  $M_2$  and  $M_3$

Define  $C^{ANALYT}$  as the  $1 \times n$  vector of analytic control variates for the  $n$  replications. To find  $C^{ANALYT}$ , first define  $r_{ij}$  as the  $i$ -th customer's interarrival time for the  $j$ -th replication of  $M_1$ . Using the replication/deletion method with a warm-up of  $l$  customers and a total of  $m$  customers simulated, estimate the expected interarrival time during the  $j$ -th replication by

$$\bar{R}_j = \frac{1}{m-l} \sum_{i=l+1}^m r_{ij} \quad j = 1, 2, \dots, n \quad (3.34)$$

Similarly define  $v_{ij}(k)$  as the service time for the  $i$ -th customer at the  $k$ -th service center during the  $j$ -th replication of  $M_1$ . Then to estimate the expected service time at the  $k$ -th service center during the  $j$ -th replication let

$$\bar{V}_j(k) = \frac{1}{m-l} \sum_{i=l+1}^m v_{ij}(k) \quad k = 1, 2, 3; j = 1, 2, \dots, n \quad (3.35)$$

Then  $C_j^{ANALYT}$ , the analytic control variate for the  $j$ -th replication is calculated as

$$C_j^{ANALYT} = \frac{1}{\frac{1}{\bar{V}_j(1)} - \frac{1}{\bar{R}_j}} + \frac{1}{\frac{1}{\bar{V}_j(2)} - \frac{1}{\bar{R}_j}} + \frac{1}{\frac{1}{\bar{V}_j(3)} - \frac{1}{\bar{R}_j}} \quad (3.36)$$

with

$$\bar{C}^{ANALYT} = \frac{1}{n} \sum_{j=1}^n C_j^{ANALYT} \quad (3.37)$$

Define  $\bar{Y}(\hat{\beta})^{ANALYT}$  as the analytically controlled estimate of expected sojourn time,

where

$$\bar{Y}(\hat{\beta})^{ANALYT} = \bar{Y}(n) - \hat{\beta}'(\bar{C}^{ANALYT} - \mu^{ANALYT}) \quad (3.38)$$

which can be computed using equation (3.15). The estimated variance and confidence interval of  $\bar{Y}(\hat{\beta})^{ANALYT}$  can be calculated using equations (3.17) and (3.19).

**3.3.3 External Control Variate (Open Queueing Network).** Application of external control variates requires a separate, similar simulation model with a known expected sojourn time,  $\mu^{EXT}$ . Call the external simulation model  $M_3$ . Using common random numbers between models  $M_1$  and  $M_3$  the estimated expected sojourn time between the two models should be highly correlated so that variance reduction can be obtained using control variate techniques.

For our example, simulation model  $M_3$  has the same characteristics and parameters as model  $M_2$  -- a series of three M/M/1 infinite capacity queues with the same interarrival and

service time distributions as  $M_2$  (see figure 1.2). Therefore

$$\mu^{EXT} = \mu^{ANALYT} = \frac{1}{\frac{1}{\mu_1} - \frac{1}{\lambda}} + \frac{1}{\frac{1}{\mu_2} - \frac{1}{\lambda}} + \frac{1}{\frac{1}{\mu_3} - \frac{1}{\lambda}} \quad (3.39)$$

Random numbers are synchronized between  $M_1$  and  $M_3$  for all arrival times and service times at each service center. Since both  $M_1$  and  $M_2$  have exponential interarrival times, customer arrival times will be the same for each simulation model. At the service centers, the same random number streams of uniformly distributed random variables between zero and one are used to generate the random service time. The streams will produce different service times since the distributions are different for each model (Weibull and exponential respectively), but will be highly correlated since both Weibull and exponential random variables are generated using the inverse transform method [17] and therefore have a one to one relationship with each uniform  $[0, 1]$  random variable generated by each stream.

To generate the control variates, let  $y_{ij}^{EXT}$  be the system sojourn time for the  $i$ -th customer during the  $j$ -th replication of external model  $M_3$ . As with model  $M_1$ ,  $m$  customer arrivals are simulated (with a warm-up period of  $l$  customers), in order to find the mean sojourn time for replication  $j$  as

$$\bar{Y}_j^{EXT} = \frac{1}{m-l} \sum_{i=l}^m y_{ij}^{EXT} \quad j = 1, 2, \dots, n \quad (3.40)$$

Then define  $C_j^{EXT}$ , the external control variate for the  $j$ -th replication, as

$$C_j^{EXT} = \bar{Y}_j^{EXT} \quad j = 1, \dots, n, \quad (3.41)$$

and

$$\bar{C}^{EXT} = \frac{1}{n} \sum_{j=1}^n C_j^{EXT} \quad (3.42)$$

Then, the externally controlled estimated sojourn time is given by

$$\bar{Y}(\hat{\beta})^{EXT} = \bar{Y}(n) - \hat{\beta}'(\bar{C}^{EXT} - \mu^{EXT}) \quad (3.43)$$

which can be computed using equation (3.15). As with the other controlled responses, the estimated variance and confidence interval can be found using equations (3.17) and (3.19).

### 3.4 Closed Queueing Network

The closed network studied has the basic form shown in figure 3.4. The network structure is the same as the network examined by Lavenberg, Welch, and Moeller (1982). An additional variation of this network, (shown in figure 3.5) will be examined.

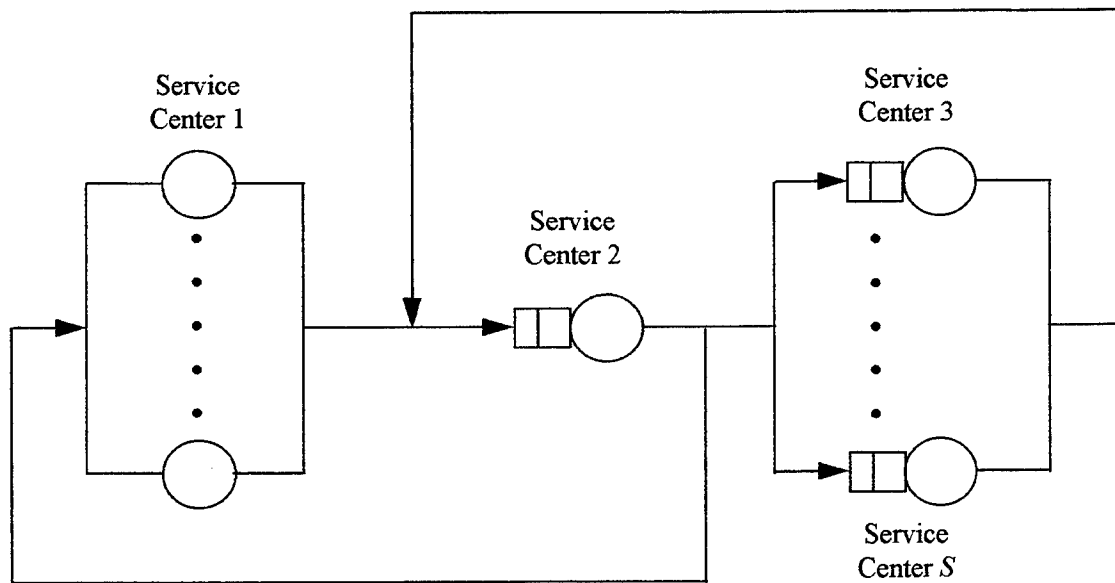


Figure 3.4. Closed queueing network model  $Q_1$

As depicted in figure 3.4, the network has a total of  $S$  service centers with a total of  $N$  customers within the closed network. Call this model  $Q_1$ . Service center 1 has exactly  $N$  servers resulting in no queues at center 1. The remainder of the centers are single server queues with customers being served with a 'first in - first out' queue priority. The underlying transition probability matrix for customer movement throughout  $Q_1$  is given by

$$P = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ p_1 & 0 & p_2 & \dots & p_S \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & 0 & \dots & 0 \end{bmatrix} \quad (3.44)$$

with  $p_k > 0$ ,  $k = 1, 3, \dots, S$ . As indicated above, all customers completing service at center 1 are routed to center 2. Upon completion of service at center 2, the customer is routed to centers 1, 3, ..., or  $S$  with probability  $p_k$ ,  $k = 1, 3, \dots, S$ .

The service time distributions at centers 1, 3, ...,  $S$  are characterized as different IID Weibull random variables with probability density function

$$f(x) = \begin{cases} \frac{\alpha}{\beta} x^{\alpha-1} e^{-x^\alpha/\beta} & \text{if } x \geq 0; \alpha, \beta > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.45)$$

Each of these centers has a service time distribution with shape parameter  $\alpha_k$  and scale parameter  $\beta_k$ ,  $k = 1, 3, \dots, S$ . Then the mean service time for each service center is equal to

$$\mu_k = \beta_k^{1/\alpha_k} \Gamma\left(\frac{\alpha_k + 1}{\alpha_k}\right) \quad k = 1, 3, \dots, S \quad (3.46)$$

and with variance given by

$$\sigma_k^2 = \beta_k^{2/\alpha_k} \left( \Gamma\left(\frac{\alpha_k + 2}{\alpha_k}\right) - \Gamma\left(\frac{\alpha_k + 1}{\alpha_k}\right)^2 \right) \quad k = 1, 3, \dots, S \quad (3.47)$$

Service center 2 has an exponential service time distribution with probability density function

$$f(x) = \begin{cases} \frac{1}{\lambda} e^{-x/\lambda} & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.48)$$

with mean service time  $\lambda$ .

$Q_1$  can be considered a simple model of an interactive multiprogrammed computer system [10]. The customers in this network are users of the system with service center 1 representing the user terminals with the a service time associated with the user's "thinking" time between system requests. Service center 2 represents the system's central processing unit (CPU) and service centers 3, ...,  $S$  model the system's mass storage units (disk, drum, tape, etc.). The service time at center 2 represents processing time for the user's task request until either the task is completed, which will cause the customer to return to center 1 and start another "think" time, or until data from a mass storage unit is required. Service times at centers 3, ...,  $S$  model the time required to transfer data from the device it represents to the main memory where it can be acted upon by the CPU (center 2).

Since users must be allocated a portion of main memory in order to access the CPU and storage devices in these types of systems, main memory limitations may not make it possible for all users to have access at the same time. This leads to the variation of  $Q_1$  discussed previously, called  $Q_C$ , as shown in figure 3.5. In  $Q_C$  at most  $N' < N$  customers can enter the subnetwork of centers 3, ...,  $S$  with service center 3 representing the CPU and centers 4, ...,  $S$  being the mass storage devices. Service center 2 holds customers until the number of customers in the subnetwork is less than  $N'$ . The service time random variables for center 2 are identically zero and there is no queueing at center 2 if the number of customers in the subnetwork is less than  $N'$ .

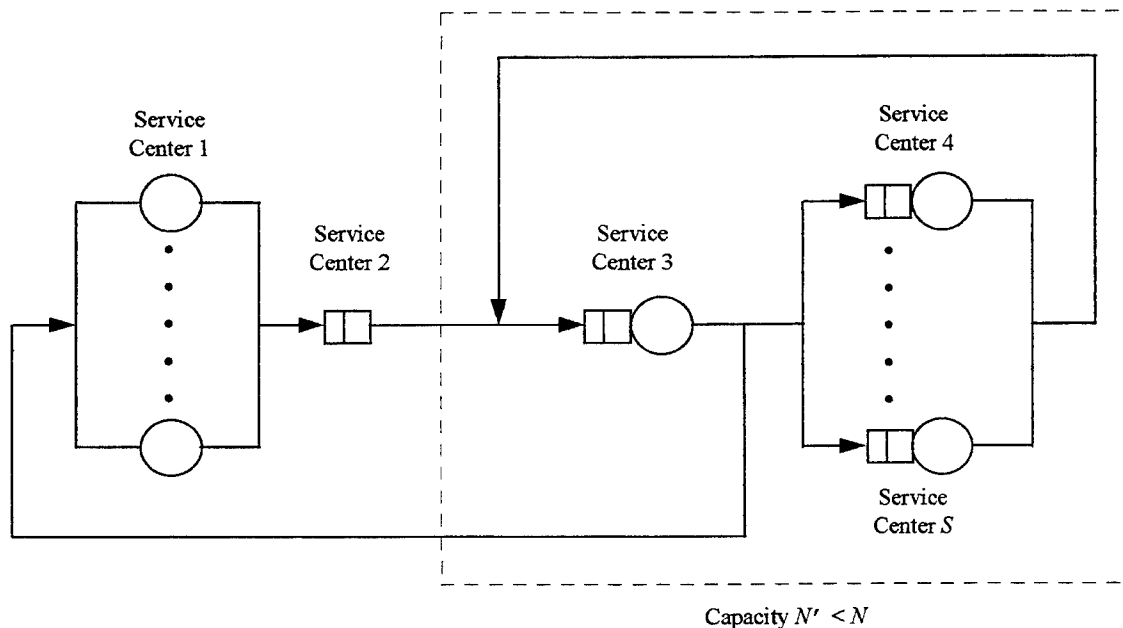


Figure 3.5. Closed queuing network model  $Q_C$

Several measurements of system performance may be of interest for such a system, but we will focus on two of them. The first being the steady state expected system response time, defined as the long run average time it takes for customers departing

service center 1 to return to center 1. The other performance measurement of interest is the steady state expected CPU utilization -- the fraction of time service center two is serving customers. It should be noted that these performance measurements are of interest in both  $Q_1$  and  $Q_C$  and the following discussion on estimating these values applies to both models.

To estimate the expected system response time for  $Q_1$ , using computer simulations, let  $\tau$  be the true steady state expected system response time which we wish to estimate, and let  $t_{ij}$  be the response time of the  $i$ -th return to center 1 during the  $j$ -th simulation replication. If we define an event as the completion of service at any of the service centers, terminate each replication upon completion of  $M$  events. Let  $Z$  be the number of events in a warm-up period in order to eliminate bias from the initial transient behavior. Define  $m_j$  as the number of times customers return to center 1 before event  $M$  and  $l_j$  as the first return to center 1 following event  $Z$  for replication  $j$ . Then let

$$T_j = \frac{1}{m_j - l_j + 1} \sum_{i=l_j}^{m_j} t_{ij} \quad j = 1, 2, \dots, n \quad (3.49)$$

so that  $\tau$  can be estimated using the replication/deletion method [13] by

$$\hat{\tau} = \bar{T}(n) = \frac{1}{n} \sum_{j=1}^n T_j \quad (3.50)$$

which has an estimated variance of

$$\frac{S_{\bar{T}}^2(n)}{n} = \frac{1}{n} \frac{\sum_{j=1}^n (T_j - \bar{T}(n))^2}{n-1} \quad (3.51)$$



and an approximate  $100(1-\alpha)$  percent confidence interval given by

$$\bar{T}(n) \pm t_{n-1, 1-\alpha/2} \sqrt{\frac{S_{\bar{T}}^2(n)}{n}} \quad (3.52)$$

CPU utilization can be estimated in the following manner. Let  $\nu$  be the true steady state expected CPU utilization. Now let  $b_{ij}$  be the amount of simulated time service center 2 is serving the  $i$ -th customer to visit center 2 during replication  $j$ . Define  $e_j$  as the simulated time of replication  $j$  when event  $M$  occurs and  $g_j$  as the simulated time of replication  $j$  when event  $Z$  occurs. Further let  $q_j$  equal the number of customers that complete service at center 2 before time  $e_j$  and let  $r_j$  be the first customer to complete service at center 2 following time  $g_j$  during replication  $j$ . Then let

$$U_j = \frac{1}{e_j - g_j} \sum_{i=r_j}^{q_j} b_{i,j} \quad j = 1, 2, \dots, n \quad (3.53)$$

so that  $\nu$  can be estimated using the replication/deletion method [13] by

$$\hat{\nu} = \bar{U}(n) = \frac{1}{n} \sum_{j=1}^n U_j \quad (3.54)$$

which has an estimated variance of

$$\frac{S_{\bar{U}}^2(n)}{n} = \frac{1}{n} \frac{\sum_{j=1}^n (U_j - \bar{U}(n))^2}{n-1} \quad (3.55)$$

and an approximate  $100(1-\alpha)$  percent confidence interval given by

$$\bar{U}(n) \pm t_{n-1, 1-\alpha/2} \sqrt{\frac{S_{\bar{U}}^2(n)}{n}} \quad (3.56)$$

3.4.1 *Internal Control Variates (Closed Queueing Network).* Possible random variables with known means for model  $Q_1$  include the  $S$  service time distributions and  $S-1$  values of the routing variables  $p_1, p_3, \dots, p_S$ . Again, the following discussion applies to  $Q_C$  as well as  $Q_1$ . As in the open queueing network, standardized work variables should be good control variate candidates. Another set of candidate internal control variates with demonstrated utility that are appropriate to  $Q_1$  are standardized routing variables [2]. Each of these internal control variates are applied to estimate both expected response time and expected CPU utilization.

The standardized work variables are derived in the same manner as they were for model  $M_1$ . Let  $s_i(k)$ ;  $i \geq 1$ ,  $k = 1, 2, \dots, S$  represent the service time process at the  $k$ -th service center. Further let  $a_j(k)$  be the number of service times completed at center  $k$ , and let  $b_j(k)$  be the first service time completed at center  $k$  after time  $g_j$  for replication  $j$ . Then the standardized work variables are

$$W_{kj} = \frac{1}{\sqrt{a_j(k) - b_j(k) + 1}} \sum_{i=b_j(k)}^{a_j(k)} \frac{s_{ij}(k) - \mu_k}{\sigma_k} \quad (3.57)$$

for  $j = 1, 2, \dots, n$  and  $k = 1, 2, \dots, S$

where  $\mu_k$  and  $\sigma_k$  are given in equations (3.46) and (3.47) respectively. Each  $W_{kj}$  has a mean of zero as  $a_j(k) - b_j(k)$  goes to infinity [23].

Standardized routing variables are developed in the following manner. Given the five branching processes at service center 2, (center 3 for  $Q_C$ ), define an indicator variable as

$$I_{ij}(k) = \begin{cases} 1 & \text{if the } i\text{-th departing customer goes to center } k \text{ for replication } j, \\ 0 & \text{otherwise.} \end{cases} \quad (3.58)$$

Then a standardized routing variable for activity  $k$  for replication  $j$  is defined as

$$R_{kj} = \sum_{i=b_j(2)}^{a_j(2)} \frac{I_{ij}(k) - p_k}{\left\{ [a_j(2) - b_j(2) + 1][1 - p_k] p_k \right\}^{1/2}} \quad (3.59)$$

for  $k = 1, 3, 4, \dots, S$  and  $j = 1, 2, \dots, n$

where  $[a_j(2) - b_j(2) + 1]$  is the total number of service completions at service center 2 for replication  $j$  during the time interval  $[g_j, e_j]$  and the  $p_k$  are the transition probabilities from service center 2 to center  $k$  as given by  $Q_1$ 's transition probability matrix  $P$ . It has been shown that the vector of routing variables,  $R$ , converges in distribution to a multivariate normal distribution with a mean of  $0$  as the simulation run length increases [2].

As with model  $M_1$ , the  $2S - 1$  internal control variates make  $2^{2S-1} - 1$  different combinations of control variates, so define  $C^{INT}$  as the  $(2S - 1) \times n$  matrix of all possible internal control variates for  $n$  replications. Then for  $p = 1, 2, \dots, 2S - 1$ ; let  $C_p^{INT}$  be one of the possible matrices of selected control variates and let  $\bar{C}_p^{INT}$  be defined as the mean vector of the associated matrix  $C_p^{INT}$  (same as equation (3.30)).

The controlled system response time,  $\bar{T}(\hat{\beta})_p^{INT}$ , is defined by

$$\bar{T}(\hat{\beta})_p^{INT} = \bar{T}(n) - \hat{\beta}'(\bar{C}_p^{INT} - 0) \quad (3.60)$$

which can be calculated using equation (3.15) and the estimated variance and confidence intervals can be found using equations (3.17) and 3(.19).

Similarly, the controlled CPU utilization fraction  $\bar{U}(\hat{\beta})_p^{INT}$  is

$$\bar{U}(\hat{\beta})_p^{INT} = \bar{U}(n) - \hat{\beta}'(\bar{C}_p^{INT} - \mathbf{0}) \quad (3.61)$$

and the associated variance and confidence intervals can be estimated using the regression equations of section 3.2.

**3.4.2 Analytical Control Variate (Closed Queueing Network).** To find analytical control variates for both  $Q_1$  and  $Q_C$  consider a new model,  $Q_2$ , that can be solved analytically and is similar in structure to models  $Q_1$  and  $Q_C$ . Let  $Q_2$  have the same structure and transition probability matrix as model  $Q_1$ , except that all service time distributions for  $Q_2$  are IID exponential random variables with mean service times equal to those of  $Q_1$ . Using the Mean Value Analysis (MVA) algorithm [11] the steady state expected system response time and CPU utilization fraction can be determined exactly. Let  $\tau^{ANALYT}$  and  $\nu^{ANALYT}$  be the expected response time and expected CPU utilization for the analytical model  $Q_2$ .

The MVA algorithm is for closed networks that have closed form solutions. The algorithm yields the mean values of the network performance characteristics response time, queue length, throughput, and utilization for each service center in the network. Basically the MVA solves the network by first determining system characteristics with only one customer in the system and then solves it for two customers based on the information obtained for one customer using the mean value theorem. The mean value

theorem relates the response time of a service center when  $n$  customers are present to the length of the queue at the service center when  $n - 1$  customers are present. The algorithm is reapplied until the system is solved for  $N$  customers.

To calculate the value of the analytical control variate for each replication, the ForQue computer program [7], which implements the MVA algorithm, is used. ForQue requires the following information about a network in order to arrive at a solution: number of customers ( $N$ ), number of service centers ( $S$ ), number of servers at each center ( $N$  at center 1, one at all the others), mean service times at each center ( $\mu_k$ ), and the transition probability matrix ( $P$ ). Obviously,  $N$ ,  $S$ , and the number of servers remain constant for each replication. Additionally, rows 1, 3, ...,  $S$  of  $P$  remain the same for each replication and are all identically equal to the  $1 \times S$  vector  $[0 \ 1 \ 0 \ \dots \ 0]$ . To find the analytical control variates the mean service times and actual branching proportions from center 2 must be calculated and input to ForQue for each replication.

Define  $C(T)_j^{ANALYT}$  as the analytical control variate for expected response time for the  $j$ -th replication. Also let  $C(U)_j^{ANALYT}$  be the analytical control variate for expected CPU utilization. To find each of these control variates using ForQue, define  $v_{ij}(k)$  as the service time of the  $i$ -th customer to visit center  $k$  during replication  $j$ . Then to find the average service time at each center let

$$V_{kj} = \frac{1}{m_j(k) - l_j(k) + 1} \sum_{i=l_j(k)}^{m_j(k)} v_{ij}(k) \quad k = 1, 2, \dots, S; \quad j = 1, 2, \dots, n \quad (3.62)$$

where  $m_j(k)$  is the number of customers completing service at center  $k$  and  $l_j(k)$  is the first customer to complete service at center  $k$  following time  $g_j$  for replication  $j$ . To find the realized routing proportions from center 2, define the indicator variable  $I_{ij}(k)$  that equals one when the  $i$ -th customer leaving center 2 enters center  $k$ , and zero otherwise for replication  $j$ . Then let

$$\hat{p}_{jk} = \frac{\sum_{i=1}^{m_j(k)} I_{ij}(k)}{\sum_{k=1}^S \sum_{i=l_j(2)}^{m_j(2)} I_{ij}(k)} \quad k = 1, 3, \dots, S \text{ and } j = 1, 2, \dots, n \quad (3.63)$$

Given these values, following each replication  $j$ , input  $V_{jk}$  for  $k = 1, 2, \dots, S$  and  $\hat{p}_{jk}$  for  $k = 1, 3, \dots, S$  into ForQue to obtain  $C(T)_j^{ANALYT}$  and  $C(U)_j^{ANALYT}$ . Then the controlled responses are

$$\bar{T}(\hat{\beta})^{ANALYT} = \bar{T}(n) - \hat{\beta}'(\bar{C}(T)^{ANALYT} - \tau^{ANALYT}) \quad (3.64)$$

and

$$\bar{U}(\hat{\beta})^{ANALYT} = \bar{U}(n) - \hat{\beta}'(\bar{C}(U)^{ANALYT} - \nu^{ANALYT}) \quad (3.65)$$

where  $\bar{C}(T)^{ANALYT}$  and  $\bar{C}(U)^{ANALYT}$  are the means of the  $n$  values of each analytical control variate. These controlled responses and their associated estimated variances and confidence intervals can be calculated using the equations of section 3.2.

**3.4.3 External Control Variate (Closed Queueing Network).** External control variates can be obtained by creating a simulation model,  $Q_3$ , of the same form as model  $Q_2$ . The true steady state expected response time,  $\tau^{EXT}$  and expected CPU utilization,  $\nu^{EXT}$ , will be the same as those for model  $Q_2$ . Using common random numbers, this

simulation model can then be used to produce external control variates for both models  $Q_1$  and  $Q_C$ . However, due to the nature of a closed network, exact synchronization of common random numbers becomes very difficult.

Since the service time distributions of centers 1, 3, ...,  $S$  in  $Q_1$  are IID Weibull random variables, it is possible to use the same underlying uniform random number to produce the  $i$ -th service time at service center  $k$  in  $Q_3$ , producing highly correlated service times. Additionally, the routing random variables for the  $i$ -th customer to complete service at service center 2 can be the same random number for both models  $Q_1$  and  $Q_2$ . The inability to produce exact synchronization with this scheme comes from the fact that the  $i$ -th service time at service center  $k$ , although highly correlated, will not be exactly the same for both models which results in a different sequence of specific customers arriving at service center 2 for each model so that the each customer will likely have a different sequence of service centers visited for each model. It seems the only way around this is to assign all routing and service time random variables to each specific customer at the beginning of each replication for both  $Q_1$  and  $Q_3$ . Although this is possible, it is an extremely cumbersome way to program both simulation models.

Although exact synchronization of common random numbers is not achieved, the following scheme for random number generation is used for  $Q_3$ . The same random number streams of uniformly distributed random numbers between zero and one will be used to generate the  $i$ -th service time exponential random variable for  $Q_3$  and Weibull random variable for  $Q_1$ , at service centers 1, 3, ...,  $S$ . The stream of routing random variables will be identical for both models as well as the stream of random service times at

service center 2. Based on this scheme, the output of  $Q_3$  should still have some correlation with  $Q_1$  and produce variance reduction.

To generate the external control variate for expected response time, let  $t_{ij}^{EXT}$  be the response time for the  $i$ -th customer for replication  $j$  of model  $Q_3$ . Using the same number of events ( $M$ ) to terminate  $Q_3$  and  $Q_1$ , and the same number of events ( $Z$ ) to define the warm-up period for both models, define  $e_j^{EXT}$  as the simulated time in which  $M$  events occur for replication  $j$  of  $Q_3$ . Further define  $g_j^{EXT}$  as the simulated time  $Z$  events occur for replication  $j$  of  $Q_3$ . Then define  $m_j^{EXT}$  as the number of customers that return to service center 1 before time  $e_j^{EXT}$  and let  $l_j^{EXT}$  be the first customer to return to center 1 following time  $g_j^{EXT}$  for replication  $j$ . Then define the external control variate for response time as

$$C(T)_j^{EXT} = \frac{1}{m_j^{EXT} - l_j^{EXT} + 1} \sum_{i=l_j^{EXT}}^{m_j^{EXT}} t_{ij}^{EXT} \quad j = 1, 2, \dots, n \quad (3.66)$$

For the external control variate for expected CPU utilization, let  $b_{ij}^{EXT}$  be the amount of time it takes to service customer  $i$  at service center 2 for replication  $j$  of model  $Q_3$ . Also let  $q_j^{EXT}$  be the number of customers that complete service at service center 2 before  $e_j^{EXT}$  and let  $r_j^{EXT}$  be the first customer to complete service at center 2 following time  $g_j^{EXT}$  for replication  $j$ . Then the external control variate for CPU utilization for replication  $j$  is defined as

$$C(U)_j^{EXT} = \frac{1}{e_j^{EXT} - g_j^{EXT}} \sum_{i=r_j^{EXT}}^{q_j^{EXT}} b_{ij}^{EXT} \quad j = 1, 2, \dots, n \quad (3.67)$$



The externally controlled responses are given by

$$T(\hat{\beta})^{EXT} = \bar{T}(n) - \hat{\beta}'(\bar{C}(T)^{EXT} - \tau^{EXT}) \quad (3.68)$$

and

$$U(\hat{\beta})^{EXT} = \bar{U}(n) - \hat{\beta}'(\bar{C}(U)^{EXT} - \nu^{EXT}) \quad (3.69)$$

where  $\bar{C}(T)^{EXT}$  and  $\bar{C}(U)^{EXT}$  are the means of the  $n$  values of the respective external control variates. These controlled responses and associated estimated variance and confidence intervals can be found using the regression formulas.

## **4. Results**

### **4.1 Introduction**

This chapter details the results of several experiments on both the open and closed queueing networks described in chapter 3. First, experimental procedures used for both networks are discussed below, followed by a discussion on computer implementation of the simulation models and the controlled response formulas. Finally, experimental results from the from the two queueing models are presented.

### **4.2 Experimental Procedures**

Several experiments are conducted on each network. Number of replications and network parameters are changed for each design point. Three different network settings have been chosen for the open network and twelve different network settings have been selected for the closed network (six for  $Q_1$  and six for  $Q_C$ ). Selection of particular network settings and their values are discussed below in section 4.2.2. At each network setting, 100 experiments are conducted with replication size of 10, and 50 experiments are conducted with replication size set at 20.

For every experimental setting, performance measures for internal, analytical, and external control variates are compared. Estimated variance, confidence interval size, coverage, and estimated Mean Square Error values for the controlled responses are compared to the same values for the uncontrolled responses. For the internal control variates, comparisons are made for all possible combinations of internal control variates for the closed network. All possible combinations of standardized work variables and

standardized routing variables are considered separately for the open network. The internal control variates are then screened and all combinations of the best performing internal control variates are considered for the closed network. Performance considerations are described below.

*4.2.1 Comparison Techniques.* Comparisons were made using the generalized method presented by Bauer and Wilson [2]. Let  $\mu$  be the expected value of the performance measurement of concern. For the  $n = 10(20)$  replications for the  $k$ -th experiment,  $k = 1, 2, \dots, m$  ( $m = 100(50)$ ), an estimate of  $\mu$  is computed. Call the estimate  $\hat{\mu}_k(l)$ , where  $l = 1$  or  $2$ . When  $l = 1$ , no controls are used and  $l = 2$  when some set of control variates are used. In a similar manner, let  $\hat{\sigma}_k^2(l)$  be the uncontrolled ( $l = 1$ ) and controlled ( $l = 2$ ) estimates, for the  $k$ -th experiment, of the variance of  $\hat{\mu}_k(l)$ . Then the average variance estimator over all  $m$  experiments for a given setting is

$$\hat{\sigma}^2(l) = \frac{1}{m} \sum_{k=1}^m \hat{\sigma}_k^2 \quad l = 1, 2 \quad (4.1)$$

The percentage change in estimated variance due to the use of control variates is then estimated by  $100 \left[ \hat{\sigma}^2(1) - \hat{\sigma}^2(2) \right] / \hat{\sigma}^2(1)$ .

For the  $k$ -th experiment the confidence interval estimate is given by

$$\hat{\Lambda}_k(l) = \hat{\mu}_k(l) \pm \hat{H}_k(l) \quad (4.2)$$

where  $\hat{H}_k(l)$  is the estimated half-width as given in equation (3.19) with  $\alpha = 0.10$ . By letting  $\hat{L}_k(l) = 2\hat{H}_k(l)$  be the estimated width of confidence interval, we can find the average width of the confidence interval estimator over all  $m$  experiments for a given

setting as

$$\hat{L}(l) = \frac{1}{m} \sum_{k=1}^m \hat{L}_k(l) \quad l = 1, 2. \quad (4.4)$$

Then, as with the variance estimates, the percentage change in estimated confidence interval width due to the use of a particular set of control variates is estimated by  $100[\hat{L}(1) - \hat{L}(2)]/\hat{L}(1)$ .

An area of concern for control variate performance is the amount of bias in the controlled estimates of  $\mu$ . Any bias induced is due to the fact that  $\beta$  must be estimated and is generally not independent of  $\bar{Y}(m)$  [13]. One measurement that is effected by bias is an estimated confidence interval coverage probability. To find one, let

$$\hat{I}_k(l) = \begin{cases} 1 & \text{if } \mu \in \hat{\Lambda}_k(l), \\ 0 & \text{otherwise} \end{cases} \quad (4.3)$$

for  $l = 1, 2$  and  $k = 1, 2, \dots, m$ . As before let  $l = 1$  for no controls and  $l = 2$  when some set of controls is used. Then an estimate of the confidence interval coverage probability is given by the actual coverage fraction for  $\hat{\Lambda}_k(l)$  given by

$$\hat{I}(l) = \frac{1}{m} \sum_{k=1}^m \hat{I}_k(l) \quad l = 1, 2 \quad (4.4)$$

Realized coverage may not always be the best indicator of bias. For example, a point estimate may be "close" to  $\mu$ , but if the associated confidence interval is small enough, coverage may not be realized. In order to measure this "closeness", another measure of bias, the estimated value of the Mean Square Error of a point estimator, will be computed. To estimate the average value of the Mean Square Error for all  $m$  experiments of a given

setting let

$$\hat{M}(l) = \frac{1}{m} \sum_{k=1}^m (\hat{\mu}_k(l) - \mu)^2 \quad l = 1, 2 \quad (4.5)$$

**4.2.2 Open Queueing Network Settings.** The open queueing network presented in section 3.3 is examined at three different settings. For a given arrival rate of 20 customers per time period, the traffic intensity at every server station is set to approximately 0.50, 0.75, and 0.90 for each experimental setting. Specific system settings for model  $M_1$  are given in table 4.1. At each of these settings, results are obtained for replication sizes of 10 and 20. For the first two design points 1,000 customers are simulated for every replication with data from the first 300 customers ignored in order to minimize the effect of the initial transient. Statistics are gathered on the final 700 customers. More customers were simulated for the third design point (0.90 traffic intensity) since it takes longer to exhibit steady-state behavior due to the saturated state of the system. In this case 20,000 customers transit through the system with data ignored on the first 10,000 customers.

Design Point	Activity	Distribution	$\alpha$	$\beta$	Mean	Variance
1.	Arrivals	exponential	~	~	20.0	400.000
	Center 1 service	Weibull	3.6150	6000.0	10.0	9.448
	Center 2 service	Weibull	3.5416	6000.0	10.5	10.808
	Center 3 service	Weibull	3.2080	3000.0	10.0	13.829
2.	Arrivals	exponential	~	~	20.0	400.000
	Center 1 service	Weibull	3.0850	6000.0	15.0	28.266
	Center 2 service	Weibull	3.0489	6000.0	15.5	30.830
	Center 3 service	Weibull	2.8355	3000.0	15.0	32.883
3.	Arrivals	exponential	~	~	20.0	400.000
	Center 1 service	Weibull	3.4398	30000.0	18.0	33.479
	Center 2 service	Weibull	3.4081	30000.0	18.5	35.958
	Center 3 service	Weibull	3.7790	3000.0	18.0	34.591

Table 4.1. Experimental Settings for open queueing network

4.2.3 *Closed Queueing Network Settings.* For the closed queueing network presented in section 3.4., six different experimental design points are selected for both models  $Q_1$  and  $Q_C$ . For all design points, the number of service stations,  $S$ , is 6 ( $S = 7$  for model  $Q_C$ ), and the number of customers,  $N$ , is equal to 25. For model  $Q_C$ , the number of customers allowed into the subnetwork,  $N'$ , is equal to 5 for all settings. To create the 6 different settings for each model, 2 different transition probability matrices are applied to 3 sets of service time distributions. By applying equation (3.43), the 2 transition probability distributions are provided in table 4.2 and the 3 service time settings are listed in table 4.3. Then the 12 different design points for the closed queueing network are outlined in table 4.4. Results are obtained for each design point for replication size of both 10 and 20.

In all 12 cases, replications of the closed queueing network are terminated following the completion of 2,000 events. To minimize the effect of the initial transient behavior, data from the first 500 events is ignored. In addition, the initial state of the network (number of customers in each service center) is based on the expected number of customers at each center for analytical model  $Q_2$ . These expected values are determined by solving the system with the ForQue program. The probability that a customer is at a given service center is determined by dividing the expected number of customers at a service center by the number of total customers (25). Then at the start of each replication (for  $Q_1$ ,  $Q_C$ , and  $Q_3$ ) each customer is assigned a uniform random number in the interval  $[0, 1]$  and is routed to a particular service center by the above probabilities.

Model $Q_1 (Q_C)$					
Matrix	$p_1$	$p_3 (p_4)$	$p_4 (p_5)$	$p_5 (p_6)$	$p_6 (p_7)$
$P_1$	0.20	0.36	0.36	0.040	0.040
$P_2$	0.25	0.30	0.30	0.075	0.075

Table 4.2. Transition probability matrix values for closed queueing network

Model $Q_1 (Q_C)$					
Service Center	Distribution	$\alpha$	$\beta$	Mean	Variance
<b>Setting A.</b>					
1	Weibull	1.46824	1000.0	100.00	4795.784
2 (3)	exponential	~	~	1.00	1.000
3 (4)	Weibull	5.6476	10.0	1.39	0.081
4 (5)	Weibull	5.64760	10.0	1.39	0.081
5 (6)	Weibull	2.61249	1000.0	12.50	26.442
6 (7)	Weibull	2.61249	1000.0	12.5	26.442
<b>Setting B.</b>					
1	Weibull	1.46824	1000.0	100.0	4795.784
2 (3)	exponential	~	~	1.0	1.000
3 (4)	Weibull	1.50438	10.0	4.17	7.973
4 (5)	Weibull	5.64760	10.0	1.39	0.081
5 (6)	Weibull	2.61249	1000.0	12.5	26.442
6 (7)	Weibull	2.61249	1000.0	12.5	26.442
<b>Setting C.</b>					
1	Weibull	1.46824	1000.0	100.0	4795.784
2 (3)	exponential	~	~	1.0	1.000
3 (4)	Weibull	5.6476	10.0	1.39	0.081
4 (5)	Weibull	5.6476	10.0	1.39	0.081
5 (6)	Weibull	2.06810	1000.0	25.0	160.798
6 (7)	Weibull	2.61249	1000.0	12.5	26.442

Table 4.3. Service time distribution settings for the closed queueing network

Design Point	Model		Transition Matrix		Service Time Setting		
	$Q_1$	$Q_C$	$P_1$	$P_2$	A.	B.	C.
1	X		X		X		
2	X		X			X	
3	X		X				X
4	X			X	X		
5	X			X		X	
6	X			X			X
7		X	X		X		
8		X	X			X	
9		X	X				X
10		X		X	X		
11		X		X		X	
12		X		X			X

Table 4.4. Design points for closed queueing network

#### 4.3 Computer Implementation

All simulation models are programmed in the simulation language SLAM II and implemented on a Sun SPARC 4 work station. Response statistics, internal control variates, and external control variates are collected and calculated using FORTRAN user-inserts with the SLAM II programs. For the open queueing network of section 3.3, analytical control variates are calculated using the FORTRAN insert. The Pascal program ForQue is used to calculate the analytical control variates for the closed network. Controlled responses, estimated variances, and confidence intervals, as well as the statistical comparisons of section 4.2.1 above, are calculated using the MATLAB computer language. SLAM II and FORTRAN source code is provided in appendix A for one design point of each network. This also includes the code necessary to generate the external control variates for that design point (models  $M_3$  and  $Q_3$ ). Appendix B contains the MATLAB script files used to calculate and compare controlled response performance.



Application of these programs at a design point is a three step process. First the model under study ( $M_1$ ,  $Q_1$ , or  $Q_C$ ) is replicated 1,000 times using the SLAM II code. Following the end of each replication, the FORTRAN insert collects the appropriate statistics and calculates the values of the response values, internal control variates, and (in the case of the open network) analytical control variates. These values are then put into a MATLAB file of formatted matrices by subroutines in the FORTRAN insert. In the case of the closed network, the appropriate statistics required by ForQue to calculate the analytical control variates are put in a text file by the FORTRAN insert. Following all 1,000 replications the text file is read by ForQue which calculates the analytical control variates for each replication and puts these values in another text file. Next, the external model ( $M_3$  or  $Q_3$ ) is replicated 1,000 times in a synchronized manner. Another FORTRAN insert collects the appropriate statistics following each replication and places the external control variates in a MATLAB formatted file. Finally, the MATLAB script file reads the appropriate MATLAB and text files and calculates uncontrolled and controlled responses and performs the necessary comparisons.

The expected values of the steady-state sojourn time for the open network and the system sojourn time and CPU utilization fraction for the closed network are estimated by making 25,000 replications at each design point. While these values are estimates, the associated .90 confidence intervals are sufficiently tight (less than 0.5% of estimated value in all cases) to make good estimates of coverage and Mean Square Error for comparison purposes.

#### 4.4 Open Queueing Network Results

Analytical control variates provide significant variance and confidence interval width reduction on the estimates of customer sojourn time at each of the 3 design points for both 10 and 20 replications. Since confidence interval width is a function of variance, and of primary concern in most simulation studies, only confidence interval width reduction is presented. Table 4.5 presents confidence interval width reduction percentages achieved for the analytical and external control variates as described in section 4.2. Results for the “best” combination of internal control variates are also included. The “best” combination is defined as the combination of internal control variates that produce the most confidence interval width reduction.

Achieved coverage percentages for uncontrolled and controlled responses are presented in table 4.6. The internal control variate results are from the same combination of internal control variates used in table 4.5. Estimated Mean Square Error values are listed in table 4.7.

Traffic Intensity	Replications	Confidence Interval Width Reduction (%)		
		Analytical control variates	External control variates	Internal control variates
0.50	10	15.5	29.7	9.3
0.50	20	19.7	32.6	13.9
0.75	10	25.5	34.4	20.2
0.75	20	28.5	37.5	24.1
0.90	10	26.2	38.7	22.0
0.90	20	28.7	41.7	25.1

Table 4.5. Confidence interval width reduction (%) for open queueing network

Traffic intensity	Replications	Coverage (%)			
		Uncontrolled	Analytical control variates	External control variates	Internal control variates
0.50	10	86	89	88	91
0.50	20	86	84	90	88
0.75	10	87	72	84	86
0.75	20	88	64	80	84
0.90	10	87	84	84	89
0.90	20	92	82	82	92

Table 4.6. Realized coverage (%) for open queueing network

Traffic intensity	Replications	Mean Square Error			
		Uncontrolled	Analytical control variates	External control variates	Internal control variates
0.50	10	0.222	0.129	0.093	0.169
0.50	20	0.132	0.063	0.037	0.076
0.75	10	16.003	12.190	6.551	10.965
0.75	20	7.641	7.274	3.141	5.770
0.90	10	360.694	185.446	126.482	173.598
0.90	20	186.145	89.724	75.283	66.380

Table 4.7. Estimated Mean Square Error for open queueing network

Several interesting results become evident from the collected data. First, in terms of confidence interval width reduction, external control variates provide the best performance, followed by analytical and then internal control variates. The superior performance of the external control variates is due primarily to the fact that the customer arrival sequence is exactly the same for  $M_1$  and  $M_2$ . Furthermore, customers then proceed through the network in the same order they enter the network and experience correlated service times in only three service centers before exiting the system. Also, the performance of all of the control variates improves as traffic intensity increases, which corresponds to the increasing value of variance of the uncontrolled response estimate.

It is apparent from the table 4.6 that analytical control variates may be inducing bias into the controlled estimate of customer sojourn time. As discussed above in section 4.2, bias can be introduced in the process of estimating  $\hat{\beta}$ . Particularly poor coverage is realized for the traffic intensity of 0.75. External control variates also indicate a possible problem with bias, but not to as great an extent as that of the analytical control variates. However, internal control variate realized coverages are all within 6 percent of the nominal.

On the other hand, estimated MSE values are not indicative of significant bias for analytical control variates. In all cases the estimated MSE for analytical control variates is less than that for the uncontrolled sojourn time estimate. It should be noted though, that MSE estimates for external and internal control variates are less than that of the analytical control variates, except for one case, indicating that analytical control variates are not

performing with the same accuracy as that of the other types. The issue of bias will be explored in more detail for the closed queueing network in the next section.

Another interesting observation is that the “best” internal control variate combination at 4 of the design points is the standardized arrival variable and at the other 2 design points the combination of the standardized arrival variable and the standardized work variable for center 2 achieves the best confidence interval width reduction. (It should be noted that the 2 design points where 2 internal control variates achieve the most variance reduction consist of 20 replications. See section 4.5.1 below for a discussion of the loss factor as a probable cause of this behavior.) As a point of reference, table 4.8 lists the confidence interval width reduction of the 5 best internal control variate combinations at the design point of traffic intensity equal to 0.75 with replication size of 20. Due to the nature of the network, it can be expected that customer arrivals should drive system performance since arrivals, being exponentially distributed, are highly variable. The Weibull service times are not as variable. The performance of the standardized arrival variable, and the implicit high correlation between it and customer sojourn time, bears out this expectation. For the open queueing network, using internal control variates provides additional insight into system performance that analytical and external control variates don’t necessarily provide.

Internal control variates	Confidence interval width	Confidence interval width reduction (%)
$W_2A$	6.71	24.1
$A$	6.74	23.7
$W_3A$	6.92	21.7
$W_2W_3A$	6.93	21.6
$W_1W_2A$	7.02	20.5

Table 4.8. Performance of 5 best internal control variate combinations for open queueing network (traffic intensity = 0.75, replications = 20)

#### 4.5 Closed Queueing Network Results

Results for the closed queueing network are presented below. A discussion of confidence interval width reduction is presented followed by a discussion on estimation of bias.

*4.5.1 Confidence Interval Width Reduction.* Analytical control variates provide significant confidence interval reduction on estimates for both system sojourn time and CPU utilization for the closed queueing network. Across the range of all design points, except for a few isolated cases, analytical control variate performance is similar to that of external and internal control variates. Confidence interval width reductions, as a percentage of the uncontrolled estimated confidence interval are provided in tables 4.9 through 4.12. System sojourn time results are listed in tables 4.9 and 4.10 while results for CPU utilization are enumerated in tables 4.11 and 4.12. Results are included for analytical control variates, external control variates, and internal control variates. The internal control variates form the combination of standardized work variables and standardized routing variables, that produces the greatest reduction in confidence interval

width. It should be noted that for one design point (when measuring CPU utilization for model  $Q_C$ ) external control variates fail to produce any variance reduction. Where this occurs, the symbol “~” is used in the table.

Of particular interest is the performance of external and analytical control variates for model  $Q_C$ . External control variates fail to provide the same level of confidence interval width reduction, particularly for CPU utilization, for model  $Q_C$  as they do for model  $Q_1$ . Recall that in model  $Q_C$ , only 5 customers at a time are allocated a portion of main memory, and therefore access to the CPU and mass storage devices (centers 3 through 7). On the other hand, model  $Q_3$  has an unconstrained subnetwork and an unlimited number of customers may access the CPU and storage devices. Certainly the variance in external control variate performance is due to this dissimilarity in model structure. This difference causes the sequence of customers visiting a particular service center to differ even more than when model  $Q_1$  is the model under study. Under these conditions, the common random numbers lose even more synchronization and the system responses from each model are not as highly correlated.

Analytical control variates, on the other hand, continue to perform at the same level for  $Q_C$ . Although the underlying model,  $Q_2$ , is also unconstrained, the model relies only on the mean responses of system parameters from  $Q_C$ . Therefore, the same conditions that cause longer sojourn times or greater CPU utilization levels in model  $Q_C$  will do the same in  $Q_2$ . Hence, correlation from replication to replication is maintained and analytical control variates continue to reduce variance for  $Q_C$ .

Model $Q_1$ - System sojourn time				
Service time setting	Transition probability matrix	Confidence interval width reduction (%) (Replications = 10(20))		
		Analytical control variate	External control variate	Internal control variates
A.	$P_1$	51.8(52.5)	51.9(52.5)	44.9(56.9)
	$P_2$	42.5(43.7)	48.5(50.4)	28.6(45.0)
B.	$P_1$	46.1(46.9)	63.0(64.2)	56.8(66.4)
	$P_2$	47.1(48.4)	56.3(57.2)	42.0(52.4)
C.	$P_1$	22.0(25.8)	52.6(55.7)	28.8(43.6)
	$P_2$	18.8(21.9)	47.6(50.8)	57.6(64.3)

Table 4.9. Confidence interval width reduction for system sojourn time for model  $Q_1$

Model $Q_C$ - System sojourn time				
Service time setting	Transition probability matrix	Confidence interval width reduction (%) (Replications = 10(20))		
		Analytical control variate	External control variate	Internal control variates
A.	$P_1$	55.8(56.9)	38.8(40.0)	41.3(49.2)
	$P_2$	50.7(52.6)	28.5(32.3)	32.2(46.0)
B.	$P_1$	60.3(61.8)	37.5(39.8)	59.8(68.0)
	$P_2$	56.8(57.2)	20.4(22.4)	40.4(46.8)
C.	$P_1$	49.3(50.1)	19.9(24.4)	37.9(49.7)
	$P_2$	55.8(57.9)	7.4(10.6)	63.3(68.6)

Table 4.10. Confidence interval width reduction for system sojourn time for model  $Q_C$



Model $Q_1$ - CPU utilization				
Service time setting	Transition probability matrix	Confidence interval width reduction (%) (Replications = 10(20))		
		Analytical control variate	External control variate	Internal control variates
A.	$P_1$	54.2(55.6)	45.0(46.8)	39.0(47.1)
	$P_2$	57.4(60.5)	47.9(50.1)	31.7(54.3)
B.	$P_1$	68.7(70.6)	61.8(63.3)	77.8(81.1)
	$P_2$	54.4(57.4)	49.8(52.0)	32.2(47.8)
C.	$P_1$	39.7(42.4)	54.0(55.8)	18.8(34.8)
	$P_2$	41.6(44.9)	49.0(51.6)	80.1(83.6)

Table 4.11. Confidence interval width reduction for CPU utilization for model  $Q_1$

Model $Q_C$ - CPU utilization				
Service time setting	Transition probability matrix	Confidence interval width reduction (%) (Replications = 10(20))		
		Analytical control variate	External control variate	Internal control variates
A.	$P_1$	37.2(38.7)	12.5(16.2)	16.6(34.9)
	$P_2$	41.2(43.3)	5.6(11.0)	9.4(30.9)
B.	$P_1$	63.8(65.4)	12.8(16.6)	59.8(65.5)
	$P_2$	29.4(32.5)	~(3.6)	16.8(32.4)
C.	$P_1$	42.2(44.3)	4.9(10.5)	30.1(42.3)
	$P_2$	68.3(69.3)	5.3(9.7)	67.3(71.3)

Table 4.12. Confidence interval width reduction for CPU utilization for model  $Q_C$

Replication length seems to influence internal control variate performance more than that of the other control variates. Analytical and external control variates produce only slightly more confidence interval width reduction when 20 replications are used versus 10 replications. However, for internal control variates, the increased number of replications produce significantly more confidence interval width reduction than when only 10 replications are used. On average, internal control variates produce 11.2 percent more confidence interval width reduction for 20 versus 10 replications across all design points. Analytical and external control variates only produce 1.9 percent and 2.7 percent more reduction for 20 replications.

The poorer performance of the internal control variates at 10 replications can be explained by the loss factor discussed in section 1.2.1. Recall that the variance of a controlled point estimate is given by

$$Var[\bar{Y}(\hat{\beta})] = \frac{n-2}{n-q-2}(1-R^2)Var[\bar{Y}] \quad (4.6)$$

where  $n$  is the number of replications,  $q$  is the number of control variates, and  $R^2$  is the coefficient of correlation between the random variable  $Y$  and the vector of control variates, and the loss factor is  $(n-2)/(n-q-2)$ . For the closed queueing network, the average number of internal control variates in the “best” combination is 3.21 for 10 replications and 4.46 for 20 replications. Calculating the loss factor for 3 control variates yields 1.6 for 10 replications and 1.2 for 20 replications. For 4 control variates the loss factor is 2 and 1.29 respectively. The loss factors for both analytical and external control variates

remain fixed at 1.14 and 1.05 for 10 and 20 replications, since each is a single control variate. Obviously, at 10 replications the loss factor has a higher detrimental effect on variance reduction for the larger combinations of internal control variates than it does on the analytical and external control variate.

As with the open queueing network, internal control variates provide additional insight into system performance at each design point. By studying the reduction achieved by the different combinations of internal control variates, critical system characteristics become apparent. For example, table 4.13 lists the confidence interval width reduction achieved by the 5 best combinations of internal control variates for CPU utilization. The design point is model  $Q_1$ , with service time setting  $C_.$ , probability transition matrix  $P_2$ , and replications equal to 20. Obviously, the proportion of customers routed to center 5, and the deviation from the mean service time at center 5 play a significant role in system performance.

Internal control variate combination	Confidence interval width reduction (%)
$W_2W_5R_5$	83.6
$W_1W_2W_5R_5$	83.4
$W_2W_5R_3R_5$	83.2
$W_2W_5R_1R_5$	83.2
$W_2W_5R_1R_3R_5$	82.6

Table 4.13. Confidence interval width reduction (%) for CPU utilization. Design point:  $Q_1$ , service time setting  $C_.$ ,  $P_2$ , and 20 replications.

The results of analytically controlled responses themselves cannot provide the same insight. However, the mean value analysis algorithm used by ForQue, the program used

to calculate the analytical control variate, provides more information about the network than just system sojourn time and CPU utilization. Given the good performance of the analytical control variates for both performance measures across all design points, the output of ForQue must be highly correlated to the output of both  $Q_1$  and  $Q_C$ . In a sense, the variance reduction performance of the analytical control variates provides validation for the analytical model's ability to predict model  $Q_1$  and  $Q_C$  behavior. For example, table 4.14 is the output from ForQue using the mean values of the system parameters of the same design point described above for table 4.13. The extremely high utilization fraction and significantly longer response time at center 5 predicted by ForQue point to the same inferences about system performance as those made in the previous section by analyzing the internal control variates.

*** FORQUE PERFORMANCE REPORT ***							
Number of Customers = 25							
Sta- tion	Nmbr Chls	Average Svc Tm	Visit Ratio	Through -put	Queue Length	Respons Time	Utiliz -ation
=====	=====	=====	=====	=====	=====	=====	=====
1	25	100.0000	1.0000	0.1317	13.1747	100.0000	13.1747
2	1	1.0000	4.0000	0.5270	1.0940	2.0759	0.5270
3	1	1.3900	1.2000	0.1581	0.2810	1.7775	0.2198
4	1	1.3900	1.2000	0.1581	0.2810	1.7775	0.2198
5	1	25.0000	0.3000	0.0395	9.2069	232.9453	0.9881
6	1	12.5000	0.3000	0.0395	0.9624	24.3501	0.4940
Sojourn time = 89.7583							

Table 4.14. Output from ForQue computer program. Network settings:  $Q_2$ , service time setting C., transition probability matrix  $P_2$ .

4.5.2 *Estimation of Bias.* Realized confidence interval coverage for uncontrolled and controlled response estimates are enumerated in tables 4.15 through 4.18. Nominal coverage is 90 percent. Due to the similarity in coverage achieved, only results for 20 replications are provided. Internal control variate coverage corresponds to the same internal control variate combinations reported above for confidence interval width reduction. As with the open queuing network, analytical control variates seem to induce bias into the controlled estimates of system sojourn time and CPU utilization. Internal and external control variate coverage results indicate little, if any, problem with bias for their controlled estimates.

Estimated Mean Square Error values are listed in tables 4.19 through 4.22. Despite the indications of bias from the realized coverage percentages, estimated MSE for the analytical control variates is substantially less than that of the uncontrolled responses in almost every case. This indicates that both  $\bar{T}(\hat{\beta})^{ANALYT}$  and  $\bar{U}(\hat{\beta})^{ANALYT}$  are more accurate, on the average, in estimating system sojourn time and CPU utilization than  $\bar{T}(20)$  and  $\bar{U}(20)$ . However, some combination of the reduced confidence interval width and real bias is preventing analytical control variates from achieving reasonable levels of realized coverage. Other indicators need to be examined to determine what is affecting the coverage values.

Model $Q_1$ - System sojourn time					
Service time setting	Transition probability matrix	Coverage (%)			
		Uncontrolled response	Analytical control variate	External control variate	Internal control variate
A.	$P_1$	92	68	90	86
	$P_2$	86	64	90	84
B.	$P_1$	86	78	94	88
	$P_2$	84	62	92	88
C.	$P_1$	94	72	82	88
	$P_2$	88	88	88	90

Table 4.15. Realized coverage for system sojourn time for model  $Q_1$  (replications = 20)

Model $Q_C$ - System sojourn time					
Service time setting	Transition probability matrix	Coverage (%)			
		Uncontrolled response	Analytical control variate	External control variate	Internal control variate
A.	$P_1$	90	66	92	80
	$P_2$	88	50	94	90
B.	$P_1$	90	78	90	94
	$P_2$	94	38	96	88
C.	$P_1$	98	52	96	96
	$P_2$	88	92	86	88

Table 4.16. Realized coverage for system sojourn time for model  $Q_C$  (replications = 20)

Model $Q_1$ - CPU utilization					
Service time setting	Transition probability matrix	Coverage (%)			
		Uncontrolled response	Analytical control variate	External control variate	Internal control variate
A.	$P_1$	90	68	92	88
	$P_2$	90	62	84	88
B.	$P_1$	88	82	80	92
	$P_2$	92	58	92	90
C.	$P_1$	92	54	82	88
	$P_2$	86	92	80	86

Table 4.17. Realized coverage for CPU utilization for model  $Q_1$  (replications = 20)

Model $Q_C$ - CPU utilization					
Service time setting	Transition probability matrix	Coverage (%)			
		Uncontrolled response	Analytical control variate	External control variate	Internal control variate
A.	$P_1$	90	88	94	86
	$P_2$	84	82	84	86
B.	$P_1$	92	92	94	88
	$P_2$	88	90	88	84
C.	$P_1$	90	68	88	96
	$P_2$	90	78	86	92

Table 4.18. Realized coverage for CPU utilization for model  $Q_C$  (replications = 20)

Model $Q_1$ - System sojourn time					
Service time setting	Transition probability matrix	Estimated Mean Square Error			
		Uncontrolled response	Analytical control variate	External control variate	Internal control variate
A.	$P_1$	3.146	1.609	0.713	0.629
	$P_2$	2.976	1.635	0.511	0.842
B.	$P_1$	22.530	6.509	1.993	2.070
	$P_2$	5.244	2.602	0.590	1.070
C.	$P_1$	5.671	5.881	1.473	2.480
	$P_2$	38.641	18.654	8.239	3.938

Table 4.19. Estimated Mean Square Error for system sojourn time for model  $Q_1$  (replications = 20)

Model $Q_C$ - System sojourn time					
Service time setting	Transition probability matrix	Estimated Mean Square Error			
		Uncontrolled response	Analytical control variate	External control variate	Internal control variate
A.	$P_1$	5.830	2.635	1.390	1.623
	$P_2$	6.387	3.876	1.730	1.413
B.	$P_1$	25.902	5.902	7.755	1.679
	$P_2$	8.584	8.006	4.579	2.356
C.	$P_1$	15.605	16.121	8.027	2.623
	$P_2$	54.500	7.247	35.259	4.264

Table 4.20. Estimated Mean Square Error for system sojourn time for model  $Q_C$  (replications = 20)



Model $Q_1$ - CPU utilization					
Service time setting	Transition probability matrix	Estimated Mean Square Error ( $* 10^{-5}$ )			
		Uncontrolled response	Analytical control variate	External control variate	Internal control variate
A.	$P_1$	6.44	3.55	2.36	2.09
	$P_2$	9.08	4.19	2.88	2.59
B.	$P_1$	10.72	1.08	1.77	0.48
	$P_2$	5.82	3.10	1.26	1.49
C.	$P_1$	18.10	17.96	5.88	7.33
	$P_2$	26.70	6.46	8.84	0.55

Table 4.21. Estimated Mean Square Error ( $* 10^{-5}$ ) for CPU utilization for model  $Q_1$  (replications = 20)

Model $Q_C$ - CPU utilization					
Service time setting	Transition probability matrix	Estimated Mean Square Error ( $* 10^{-5}$ )			
		Uncontrolled response	Analytical control variate	External control variate	Internal control variate
A.	$P_1$	4.82	2.61	3.18	2.72
	$P_2$	11.10	3.86	9.02	3.84
B.	$P_1$	7.35	0.66	4.11	0.70
	$P_2$	5.25	2.15	4.81	2.37
C.	$P_1$	18.10	12.90	14.70	3.66
	$P_2$	18.60	3.11	13.20	1.77

Table 4.22. Estimated Mean Square Error ( $* 10^{-5}$ ) for CPU utilization for model  $Q_C$  (replications = 20)

Another indication of the coverage problem for analytical control variates is presented in tables 4.23 and 4.24. These tables list the estimated mean for each design point found using 25,000 replications and the average controlled response estimated using analytical control variates ( $\bar{T}(\hat{\beta})^{ANALYT}$  and  $\bar{U}(\hat{\beta})^{ANALYT}$ ). For system sojourn time, the analytically controlled response is lower than the estimated response at every design point. The situation is reversed for CPU utilization where the analytically controlled estimate is consistently higher, with the exception of 2 design points. Both of these points correspond to the same service time setting (C.) and transition probability matrix ( $P_2$ ).

Tables 4.25 and 4.26 compare the realized estimates for system sojourn time and CPU utilization with the mean values obtained at each design point using ForQue. Note that the mean value analysis of the exponential network provided by ForQue is consistently higher for system sojourn time and consistently lower for that of CPU utilization than the estimates obtained with 25,000 replications for  $Q_1$ . These relationships are directly opposite of that for the analytically controlled response and the estimated means. These relationships could have some bearing on the problem.

System sojourn time					
Model	Service time setting	Transition probability matrix	$\bar{T}(25,000)$	$\bar{T}(\hat{\beta})^{ANALYT}$	Percentage difference
$Q_1$	A.	$P_1$	38.9	37.9	-2.6
		$P_2$	32.9	31.9	-3.0
	B.	$P_1$	88.5	87.1	-1.6
		$P_2$	46.3	44.9	-3.0
	C.	$P_1$	47.8	46.0	-3.8
		$P_2$	87.7	86.9	-0.9
$Q_c$	A.	$P_1$	45.0	43.6	-3.1
		$P_2$	41.9	40.2	-4.1
	B.	$P_1$	103.9	102.2	-1.6
		$P_2$	64.1	61.6	-3.9
	C.	$P_1$	62.8	59.2	-5.7
		$P_2$	97.3	96.2	-1.1

Table 4.23. Estimated system sojourn time for 25,000 replication vs analytical controlled response for 20 replications

CPU utilization					
Model	Service time setting	Transition probability matrix	$\bar{U}(25,000)$	$\bar{U}(\hat{\beta})^{ANALYT}$	Percentage difference
$Q_1$	A.	$P_1$	0.903	0.907	0.44
		$P_2$	0.755	0.761	0.79
	B.	$P_1$	0.663	0.664	0.15
		$P_2$	0.686	0.691	0.73
	C.	$P_1$	0.850	0.861	1.29
		$P_2$	0.539	0.536	-0.56
$Q_c$	A.	$P_1$	0.866	0.868	0.23
		$P_2$	0.708	0.712	0.56
	B.	$P_1$	0.618	0.618	0.00
		$P_2$	0.612	0.615	0.49
	C.	$P_1$	0.775	0.784	3.74
		$P_2$	0.516	0.512	-0.77

Table 4.24. Estimated CPU utilization for 25,000 replication vs analytical controlled response for 20 replications

System sojourn time				
Model	Service time setting	Transition probability matrix	$\bar{T}(25,000)$	Mean value analysis (ForQue)
$Q_1$	A.	$P_1$	38.9	41.2
		$P_2$	32.9	37.2
	B.	$P_1$	88.5	91.1
		$P_2$	46.3	51.0
	C.	$P_1$	47.8	52.3
		$P_2$	87.7	89.8

Table 4.25. Comparison of estimated system sojourn time and mean value analysis

CPU utilization				
Model	Service time setting	Transition probability matrix	$\bar{U}(25,000)$	Mean value analysis (ForQue)
$Q_1$	A.	$P_1$	0.903	0.885
		$P_2$	0.755	0.729
	B.	$P_1$	0.663	0.654
		$P_2$	0.686	0.662
	C.	$P_1$	0.850	0.821
		$P_2$	0.539	0.527

Table 4.26. Comparison of estimated CPU utilization and mean value analysis

To get a better idea of exactly how this induced bias is affecting the analytically controlled estimates, the following figures are provided. Each figure depicts the estimated responses and associated confidence intervals for all experiments at a single design point. For system sojourn time, model  $Q_1$  with service time setting B., transition probability matrix  $P_2$ , and 20 replications (50 experiments) is presented. The 50 diamond symbols in each figure represent the 50 point estimates made at this design point using 20 replications. The bracketed lines above and below each diamond represent the width of

the estimated confidence interval for the associated 20 replication design point. For reference the estimated mean found using 25,000 replications is represented by the horizontal line across each figure. Figure 4.1 presents the values obtained by  $\bar{T}(20)$  and figures 4.2 and 4.3 compare  $\bar{T}(\hat{\beta})^{ANALYT}$  and  $\bar{T}(\hat{\beta})^{EXT}$  for the same experiments. Figure 4.2 clearly shows how  $\bar{T}(\hat{\beta})^{ANALYT}$  consistently underestimates system sojourn time while  $\bar{T}(\hat{\beta})^{EXT}$  appears to be evenly scattered about the estimated mean. Although the analytically controlled estimates are mostly lower than the mean, on average they are closer than the uncontrolled estimate, hence the lower estimated MSE.  $\bar{T}(\hat{\beta})^{EXT}$  on the other hand, appears to exhibit no evidence of bias.

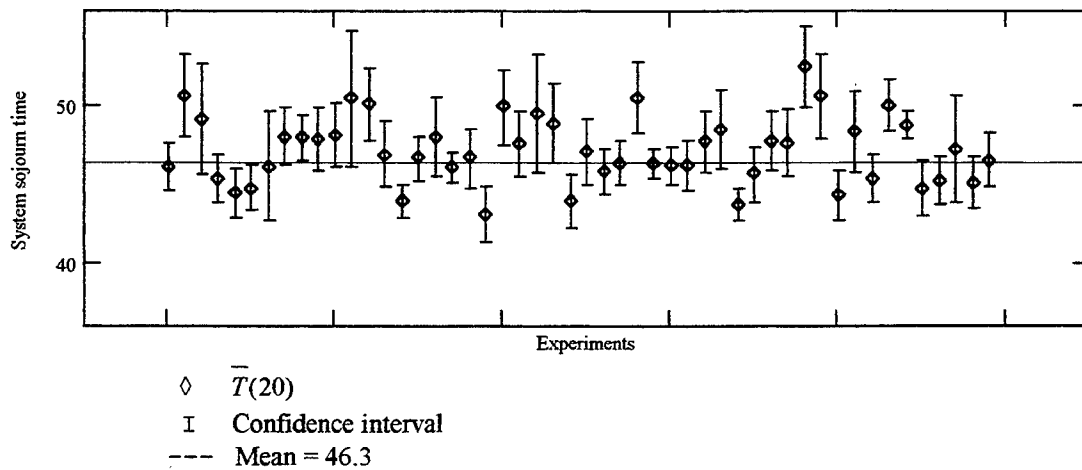


Figure 4.1  $\bar{T}(20)$  experimental results with estimated confidence interval. Design point:  $Q_1$ , service time setting B., transition probability matrix  $P_2$ . (Each point represents one point estimate and confidence interval realization.)

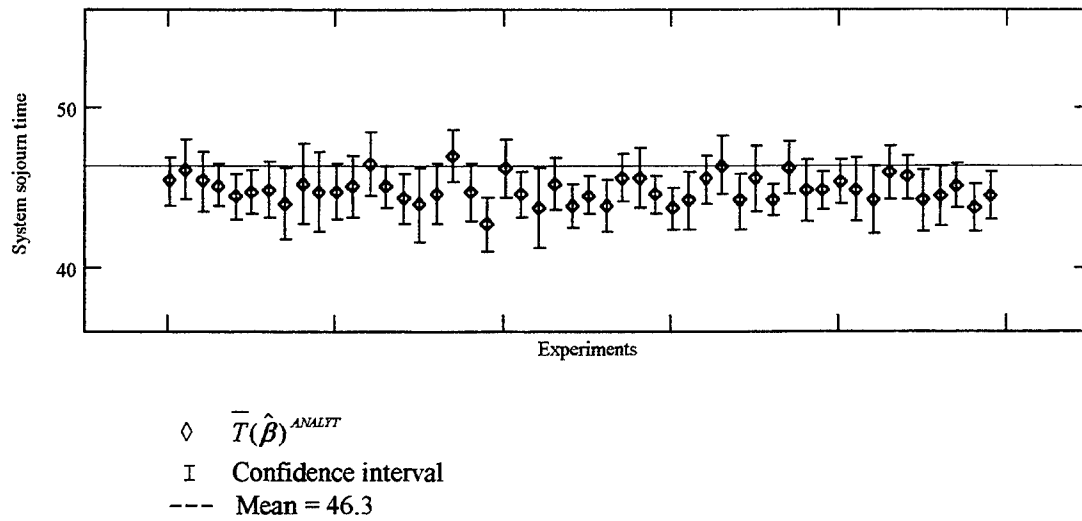


Figure 4.2.  $\bar{T}(\hat{\beta})^{ANALYT}$  experimental results with estimated confidence interval. Design point:  $Q_1$ , service time setting B., transition probability matrix  $P_2$ . (Each point represents one point estimate and confidence interval realization.)

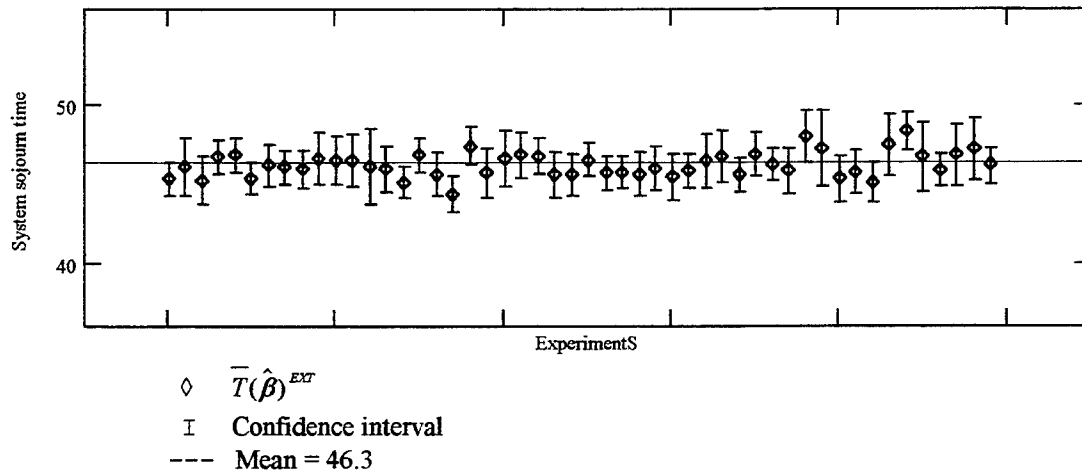


Figure 4.3.  $\bar{T}(\hat{\beta})^{EXT}$  experimental results with estimated confidence interval. Design point:  $Q_1$ , service time setting B., transition probability matrix  $P_2$ . (Each point represents one point estimate and confidence interval realization.)

Finally, model  $Q_C$  with service time setting  $C$ ., probability transition matrix  $P_2$ , and 20 replications is used to create the same figures for CPU utilization. Figure 4.4 shows the experimental results for  $\bar{U}(20)$ , and figures 4.5 and 4.6 present the experimental results for  $\bar{U}(\hat{\beta})^{ANALYT}$  and  $\bar{U}(\hat{\beta})^{INT}$ . The internal control variates used are the same as reported above for confidence interval width reduction. As above, the analytically controlled response is consistently on one side of the estimated mean while consistently closer to the mean than the uncontrolled response. Similarly, the internal control variates are evenly scattered about the mean and exhibit no evidence of bias.

The above figures coupled with the data on realized coverage and the consistent difference between the estimated mean and analytically controlled responses point to a real problem with bias for the analytical control variates in this study. The mechanism(s) causing the bias in the analytically controlled response estimates is(are) not totally clear. Means of possibly reducing the bias are discussed in the following chapter.

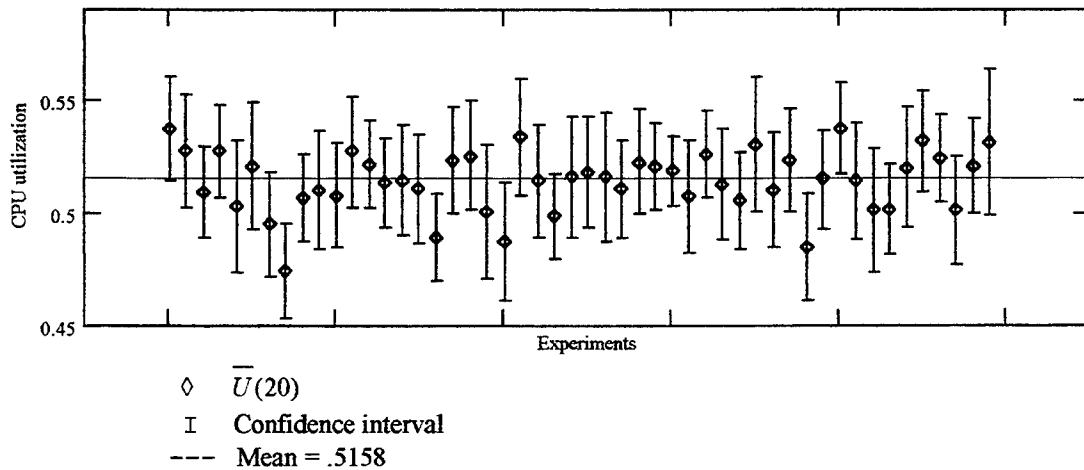


Figure 4.4.  $\bar{U}(20)$  experimental results with estimated confidence interval. Design point:  $Q_C$ , service time setting  $C$ ., transition probability matrix  $P_2$ . (Each point represents one point estimate and confidence interval realization.)

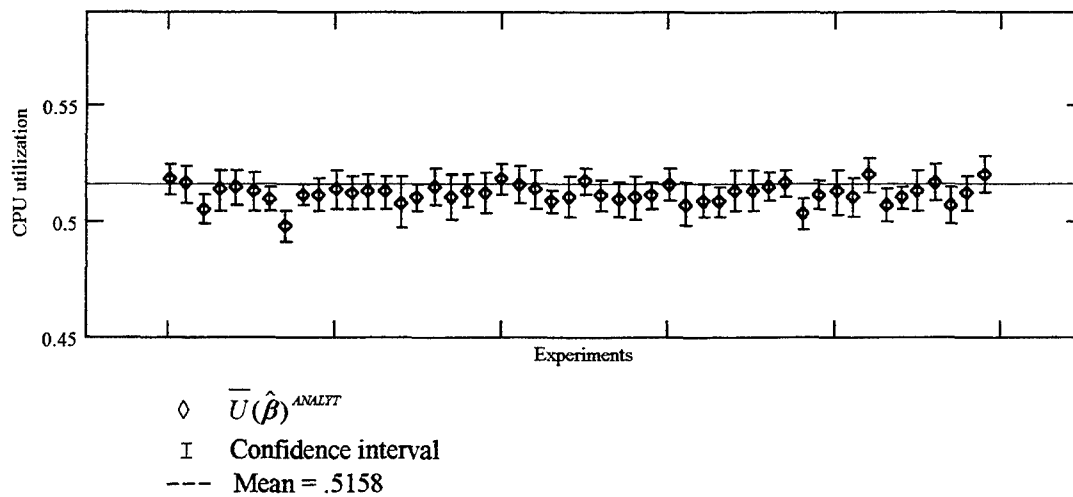


Figure 4.5.  $\bar{U}(\hat{\beta})^{ANALYT}$  experimental results with estimated confidence interval. Design point:  $Q_C$ , service time setting  $C$ ., transition probability matrix  $P_2$ . (Each point represents one point estimate and confidence interval realization.)

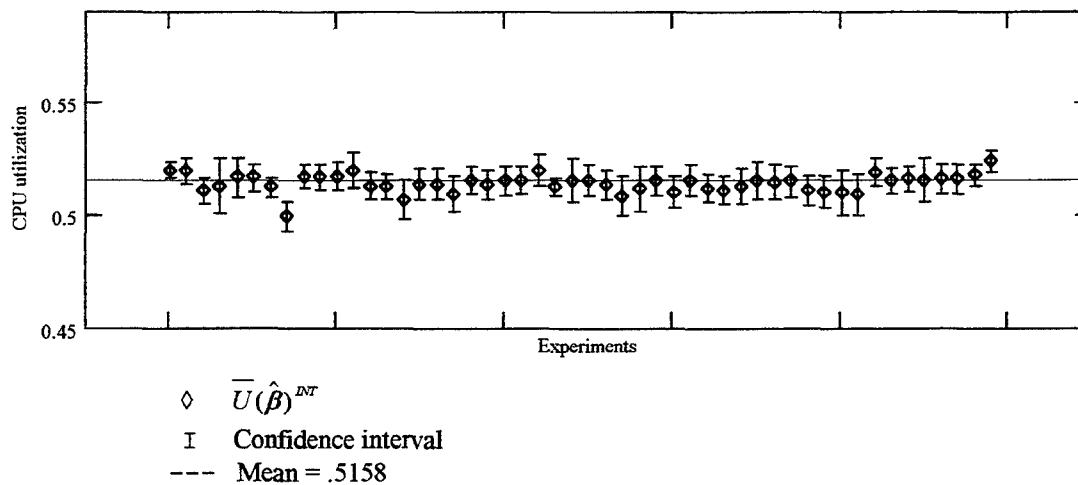


Figure 4.6.  $\bar{U}(\hat{\beta})^{INT}$  experimental results with estimated confidence interval. Design point:  $Q_C$ , service time setting  $C$ ., transition probability matrix  $P_2$ . (Each point represents one point estimate and confidence interval realization.)



## ***5. Conclusions and Recommendations for Future Research***

### ***5.1 Introduction***

The motivation for this research was to determine if a hybrid type of control variate, called an analytical control variate in this study, could effectively reduce the width of point estimate confidence intervals of replicative simulation studies while avoiding some of the limitations of internal and external control variates. In terms of confidence interval width reduction, the experimental results indicate that analytical control variates are quite successful for the networks studied. Unfortunately point estimates found using analytical control variates appear to have significant levels of bias. However, there may be occasions where the bias is not a problem. For example, analytical control variates could be useful when assessing the percentage change in performance when system parameters are changed. The following sections contain observations on the differences in application of analytical control variates versus external and internal control variates. Additionally, a discussion on possible remedies to the bias problem is presented. Recommendations for future research end this chapter.

### ***5.2 Study Time Efficiencies***

Analytical control variates outperform external control variates in this study in terms of time efficiency. For example, to generate the external control variates for the 1,000 replications at one design point, 1,000 replications of the external simulation model must be made as well. The similarity of the two models translates into approximately equal computer run times for each model. Of course, in some computers, such as the Sun

SPARC 4 work stations used in this study, these replications can be run at the same time. However parallel processing of this type increases average run time for each model. Obviously, only truly significant gains in variance reduction over other control variates would make external control variates worth while. If there is enough time to make the additional external simulations, there is time to make that many more replications of the original model and apply internal or analytical control variates to the additional replications.

Computer run times for analytical control variates are virtually instantaneous in comparison. For the open system studied, analytical control variates are calculated in the FORTRAN routine used to collect replication statistics, so analytical control variates require only a few simple calculations following each replication. Approximate run times for the closed queueing system studied are 15 minutes for 1,000 replications on a Sun SPARC 4 work station. The ForQue computer program can generate the 1,000 analytical control variates for these replications in less than 10 seconds on the same computer system.

The generation of internal control variates is virtually instantaneous as well. In both models studied, the standardized arrival, work, and routing variables are calculated in the same FORTRAN routine that collects the statistics necessary to compute them. Although the generation of the internal control variates is as rapid as that for analytical control variates, application of internal control variates may not be as quick.

Unlike analytical (and external) control variates, some effort (and time) must be spent in choosing the "best" subset of all possible internal control variates. First of all, not

all of the internal control variates generated are sufficiently correlated with the system output to produce any variance reduction. Secondly, as pointed out in section 1.2.1, at some point adding additional control variates that are correlated with the system output cannot overcome the effect of the loss factor in the estimated variance. Research has been conducted on the best way to select an optimum, or close to optimum, set of internal control variates. Two approaches are presented in Nelson (1987) (see section 2.2.2). One approach involves step-wise regression and the other proposes the computation of a marginal improvement ratio. Bauer and Wilson (1992) develop a method for multiresponse simulations that minimizes the mean-square confidence-region volume.

One obvious time advantage to internal control variates over both analytical and external control variates is that no analytical model that approximates the behavior of the system under study is necessary. Depending on the system simulated finding or creating an appropriate analytical model can be a daunting problem, however there are many useful analytical models in existence and research continues in the development of more. The mean value analysis algorithm used for the closed network in this study is just one example.

From this study, it appears that if an adequate analytical model is available, analytical control variates provide an advantage in terms of study completion times. Further, if a particular simulation model is used regularly to conduct many studies, time spent developing or finding an appropriate analytical model could pay time dividends in the future. Otherwise, internal control variates are the only option available.

### *5.3 Effectiveness over Different Model Configurations*

As the synchronization of common random numbers between the model under study and the external model decreased, the effectiveness of the external control variates decreased. Obviously, the correlation between the outputs of the two models depends on the correlation of the input random variables. As models increase in complexity, synchronicity becomes problematic. Furthermore, the service time distributions for the models under study in this research effort (Weibull) have the characteristic of being generated using the inverse transform method. This produces a one to one relationship between the Weibull random variable and the uniform random number stream used to generate them. This characteristic made the level of synchronization achieved in this study possible. Many other random variable distributions that may be required to adequately model some other situation can't be generated using the inverse transform method and therefore require several random numbers to generate one random variable [13]. Achieving any random number synchronization under those circumstances becomes extremely difficult and can increase computer run time as well.

Analytical control variates rely only on the correlation between one replication of the model under study and the result of an analytical model calculation using the observed mean values of the input random variables for that replication. As long as the analytical model responds in a correlated manner, variance reduction occurs. In this research effort, analytical control variates perform essentially the same for all models and design points considered irregardless of their departure from the analytical model. Both Tomick (1988) and Dietz and Harmonosky (1989) reported similar results.

Internal control variates don't depend on some external approximation to the model under study and model configuration had little effect on their performance. Traffic intensities and utilization rates seemed to have the most effect on internal control variate performance. Variance reduction increases, in this study, for internal control variates when the model under study has one or more service centers with high traffic intensity or utilization rates inducing high correlation between the internal control variates that correspond to that service center and the system output.

#### *5.4 System Insight*

As discussed previously in section 4.5, the variance reduction achieved for the different internal control variates provides insight into the behavior of the network under study. The internal control variates that produce the most variance reduction indicate that behavior (service times, routing probabilities) at the service centers associated with them has the highest correlation to system performance.

External and analytical control variates can't provide this same post-simulation analysis. However, the analytical models used to produce the analytical control variates can predict similar results if the model provides system parameter values (see section 4.5). Obviously these predictions are only as good as the analytical model approximates the system under study behavior. As discussed in section 4.5, the variance reduction produced by a particular analytical control variate is a function of the correlation between the analytical model and the system under study. Significant variance reduction across a range of system parameters indicates that the analytical model output tracks with the

output of the model under study. Given this variance reduction validation, the analytical model could be used by analysts to quickly provide decision makers with useful answers when there is not enough time to make simulation studies. The analytical models could be used in much the same way response surface meta-models are now used.

Analytical control variates can also provide pre-simulation analysis that neither of the other two types of control variates can. The analytical model used to generate analytical control variates can also be used to find excellent starting points for studies when different configurations or resources are being compared. Neither external nor internal control variates can provide the analyst with the same information before any replications are made. Given a good starting point, study times for large projects can be significantly reduced.

### *5.5 Loss Factor and Confidence Interval Width*

Discussion in Chapter 1 pointed out that the variance estimator when control variates are used includes a loss factor of  $(n-2)/(n-q-2)$  where  $n$  is the number of replications and  $q$  is number of control variates. Further, the degrees of freedom for the associated  $t$ -statistic used to estimate a confidence interval is equal to  $n-q-1$ . Of course, if an additional control variate is sufficiently correlated to the response statistic, its addition can overcome the loss factor and reduced degrees of freedom. Replication size must also be considered when taking the loss factor into account. For example, in this study, internal control variates show higher levels of improvement in confidence interval width reduction over the other control variates when replications are increased from 10 to 20 (see section 4.5.1).

However, even at 10 replications, internal control variates performed at similar levels of variance reduction as the other control variates. It is this problem that leads to the process of internal control variate selection discussed above in section 5.2. Analytical and external control variates have fixed loss factors and associated degrees of freedom of  $(n-2)/(n-3)$  and  $n-2$  respectively. Despite this possible disadvantage for internal control variates, when some effort is made to find the best subsets, confidence interval width reduction is comparable with that of the other control variates in this study.

### *5.6 Bias of Analytical Control Variate Point Estimates*

Both this study and Tomick (1988) reveal a problem with bias in the estimation of system responses when using analytical control variates. This bias is induced when the optimal control coefficient  $\hat{\beta}$  is estimated [13]. For the regression method of estimating  $\hat{\beta}$ , we assume that the response variable and control variate(s) have a multivariate normal distribution [12]. If this is not the case, bias could be the outcome when making point estimates of the response variable.

To investigate the normality assumption, a histogram of the 1,000 analytical control variates generated to estimate system sojourn time for model  $Q_1$  with service time setting A. and transition probability matrix  $P_2$  is provided by figure 5.1 below. From the histogram, it appears the analytical control variates may have more values to the right of the mean than might be expected for a normal distribution. This additional weight in the higher value tail could be the cause for the consistently under estimating of mean sojourn time by the analytical control variates. A chi-square Goodness of Fit test, with a null

hypothesis of normality, provides a test statistic with a p-value of approximately .28. This value is insufficient to reject the null hypothesis, however distribution selection software indicates other distributions which provide a better fit of the data. In terms of the chi-square test, the lognormal, chi-square, logistic, and Erlang distributions all provide significantly better test statistics. Other design points provide similar results. Given an indication of non-normality for the analytical control variates, one possible means of reducing the induced bias is to perform a transformation on the analytical control variates that will produce values that are more nearly normally distributed.

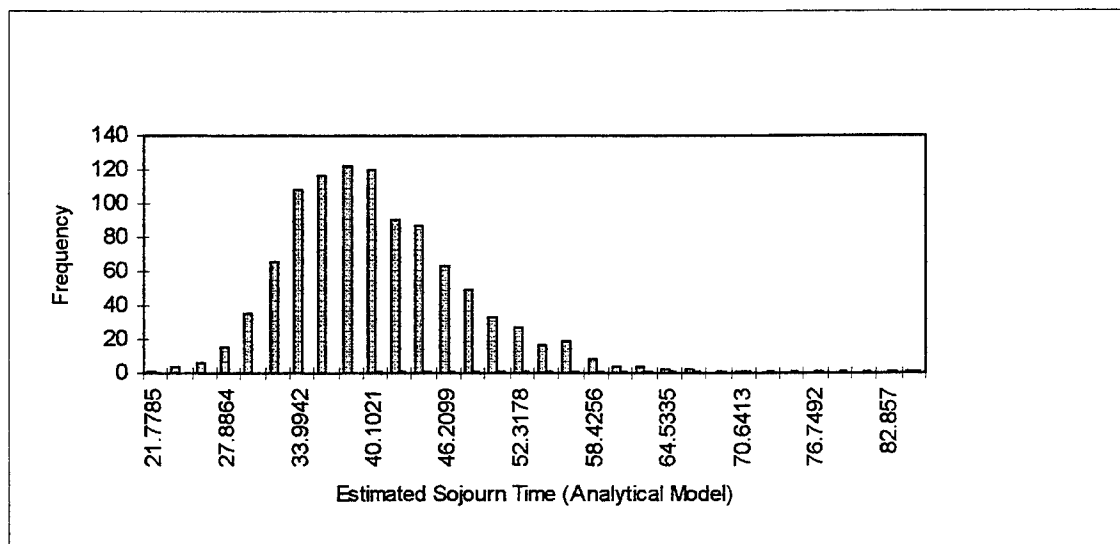


Figure 5.1. Histogram of 1,000 analytical control variate values of estimated sojourn time, for model  $Q_1$  at service time setting A. and transition probability matrix  $P_2$ .

Another means of possibly reducing the bias, suggested by Dietz [4], is to adjust the means of the analytical model parameters so that the output of the analytical model is closer to the output of the simulation model. Experimental results (see tables 4.23-4.26) show a consistently longer estimate of system sojourn time and a consistently smaller



CPU utilization rate for the analytical model versus the simulation result. In this study, the selection of the mean for the exponentially distributed service times of the analytical model is arbitrarily set to equal the mean of the simulation Wiebull service time distributions. Dietz proposes making the adjustment to the exponential distributions based on the difference between the analytical mean queueing length and an estimate of the true average queueing length. This method could be accomplished in our case by using some pilot runs of the simulation model to make the appropriate adjustments. Although this method will produce analytical solutions closer to the simulation model, it doesn't guarantee the elimination of bias.

Nelson (1990) evaluates several techniques for computing the controlled response in an effort to remedy bias problems. For the replication sizes used in this study, Nelson recommends using a technique known as splitting for estimating the response variable and associated confidence interval when bias is suspected. Simply put, splitting consists of computing  $\hat{\beta}$  from a preliminary, or pilot, sample rather than the 10(20) replications used to estimate the responses. Then even if the response variable and control variate(s) are not normal, the controlled estimate will still be unbiased. To see this, let  $\hat{\beta}^*$  be the usual regression estimator of  $\beta$  for an independent sample of  $m$  replications for response variable  $Y$  and control variate vector  $C$ . Then for  $n$  different (independent) replications, using equation (3.6), the control variate estimator

$$\bar{Y}(\hat{\beta}^*) = \bar{Y}(n) - \hat{\beta}^{*'} (\bar{C} - \mu_c) \quad (5.1)$$

is unbiased since  $\hat{\beta}^*$  and  $\bar{C}$  are independent of each other.

Nelson proposes the use of what he calls an extreme form of splitting where  $n$  replications are split into  $n$  groups. In effect this consists of estimating a different  $\hat{\beta}$  for every  $j$ -th replication from the  $n-1$  other replications. He goes on to provide computational formulas to find the controlled, split point estimates, and estimated variance and confidence interval. Nelson reports excellent results using this splitting technique with variance estimates nearly identical to those using the regression method.

### *5.7 Recommendations for Future Research*

There are many possible benefits from using analytical control variates. Before we can take advantage of them in many real applications, a solution to the bias problem must be found. Further research into the actual distribution of the analytical control variates and associated transformation techniques could provide significant reduction in the observed bias. The results from Nelson (1990) indicate future research into the application of the splitting technique with analytical control variates should be pursued. Additionally, the method of "correcting" the analytical response by adjusting the parameters of the analytical model should be explored. A study that explores the results from each of these methods separately and together should prove to be very valuable.

If any of these techniques eliminates the bias problem, research into real world applications would be appropriate. One such example that could use the same mean value analysis program ForQue would be an Air Force manpower requirement study. Studies of this type are conducted using the Logistics Composites Model (LCOM) computer simulation model. Jenkins (1994) developed a mean value heuristic for the analysis of the

aircraft sortie generation process, the same process used to evaluate manpower requirements. These studies showed excellent correlation between the output of LCOM and the heuristic (contained in ForQue) for a simple sortie generation process. This heuristic should serve as an excellent analytical model for generating analytical control variates for a manpower study.

Another area for future research is to find methods to use the variance reduction obtained from analytical control variates to validate the analytical model as an adequate surrogate model for the simulation model. A technique might be developed similar to that used to create response surface methodology meta-models.

## Appendix A: Computer Source Code

### A.1 SLAM II Source Code for Model M<sub>1</sub>

```
*****
; XX(I) Variable definitions
*****
; XX(1)=service time one distribution beta parameter
; XX(2)=service time one distribution alpha parameter
; XX(3)=service time two distribution beta parameter
; XX(4)=service time two distribution alpha parameter
; XX(5)=service time three distribution beta parameter
; XX(6)=service time three distribution alpha parameter
; XX(7)=mean of service time one distribution
; XX(8)=mean of service time two distribution
; XX(9)=mean of service time three distribution
; XX(10)=standardized work variable one
; XX(11)=standardized work variable two
; XX(12)=standardized work variable three
; XX(13)=standardized arrival variable
; XX(14)=sum of standardized work variable one
; XX(15)=sum of standardized work variable two
; XX(16)=sum of standardized work variable three
; XX(17)=sum of standardized arrival variable
; XX(18)=arrival time of previous customer
; XX(25)=counter of customers
; XX(26)=standard deviation of service time one distribution
; XX(27)=standard deviation of service time two distribution
; XX(28)=standard deviation of service time three distribution
*****
; Atrib definitions
*****
; Atrib(1)=arrival time
; Atrib(2)=center one service time
; Atrib(3)=center two service time
; Atrib(4)=center three service time
; Atrib(5)=interarrival time
; Atrib(6)=sojourn time
*****

GEN,IRISH,MICKEY D TEST MODEL,10/14/1995,1000,Y,N,Y/Y,N,N/1,132;
LIMITS,3,6,100;
INTLC,XX(1)=6000,XX(2)=3.085,XX(3)=6000,XX(4)=3.0489;
INTLC,XX(5)=3000,XX(6)=2.8355,XX(7)=15,XX(8)=15.5,XX(9)=15;
INTLC,XX(10)=0,XX(11)=0,XX(12)=0,XX(13)=0,XX(14)=0,XX(15)=0;
INTLC,XX(16)=0,XX(17)=0,XX(18)=0,XX(25)=0;
INTLC,XX(26)=5.3166,XX(27)=5.5524,XX(28)=5.7344;
NETWORK;

;
    CREATE,EXPON(20,1),,1,1000,1;
    ACTIVITY;

;
; *Assign service times*
;
ASN1  ASSIGN,ATRIB(2)=WEIBL(XX(1),XX(2),2),ATRIB(3)=WEIBL(XX(3),
        XX(4),3),ATRIB(4)=WEIBL(XX(5),XX(6),4),1;
        ACTIVITY,,,ASN2;
```

```

;
ASN2  ASSIGN, ATRIB(5)=TNOW-XX(18), XX(18)=TNOW, 1;
      ACTIVITY,,, ORDER;
;
ORDER QUEUE(1),,,,;                               Service center one
      ACTIVITY(1)/1, ATRIB(2),, PAY;
;
PAY   QUEUE(2),,,,;                               Service center two
      ACTIVITY(1)/2, ATRIB(3),, PU;
;
PU    QUEUE(3),, 2, BLOCK;                          Service center three
      ACTIVITY(1)/3, ATRIB(4),, ASN3;
;
; *Calc sojourn time and count customers*
;
ASN3  ASSIGN, ATRIB(6)=TNOW-ATRIB(1), XX(25)=XX(25) +1, 1;
      ACTIVITY,, XX(25).LE.300, TERM;                Warm-up period
      ACTIVITY,,, ASN4;
;
; *Calculate internal cv's*
;
ASN4  ASSIGN, XX(10)=ATRIB(2)/XX(26)-XX(7)/XX(26), XX(11)=ATRIB(3)/XX(27)-
      XX(8)/XX(27), XX(12)=ATRIB(4)/XX(28)-XX(9)/XX(28),
      XX(13)=ATRIB(5)/20.0-1.0;
;
      ACTIVITY,,, ASN5;
;
; *Sum internal cv's*
;
ASN5  ASSIGN, XX(14)=XX(14)+XX(10), XX(15)=XX(15)+XX(11),
      XX(16)=XX(16)+XX(12), XX(17)=XX(17)+XX(13);
      ACTIVITY,,, COL1;
;
; *Collect statistics*
;
COL1  COLCT(1), ATRIB(2), ORDERTIME,, 1;
      ACTIVITY,,, COL2;
COL2  COLCT(2), ATRIB(3), PAYTIME,, 1;
      ACTIVITY,,, COL3;
COL3  COLCT(3), ATRIB(4), PU TIME,, 1;
      ACTIVITY,,, COL4;
COL4  COLCT(4), ATRIB(5), INTER ARRIVE,, 1;
      ACTIVITY,,, COL5;
COL5  COLCT(5), ATRIB(6), SOJOURN,, 1;
      ACTIVITY,,, COL6;
COL6  COLCT(6), XX(10), WORK VAR1,, 1;
      ACTIVITY,,, COL7;
COL7  COLCT(7), XX(11), WORK VAR2,, 1;
      ACTIVITY,,, COL8;
COL8  COLCT(8), XX(12), WORK VAR3,, 1;
      ACTIVITY,,, COL9;
COL9  COLCT(9), XX(13), STD ARRV,, 1;
      ACTIVITY,,, TERM;
TERM  TERMINATE;
      END;
FIN;

```

## A.2 FORTRAN Subroutine for Model M<sub>1</sub>

```
*****
*
* Main program
*
*****

      PROGRAM MAIN
      DIMENSION NSET(5000)
      INCLUDE 'PARAM.INC'
      COMMON/SCOM1/ATRB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, II, MFA,
      1MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
      2SSL(MEQT),TNEXT, TNOW, XX(MMXXV)
      COMMON QSET(5000)
      EQUIVALENCE (NSET(1),QSET(1))
      NNSET=5000
      NCRDR=5
      NPRNT=6
      NTAPE=7
      OPEN(UNIT=NCRDR,FILE='fort.5')
      OPEN(UNIT=NPRNT,FILE='fort.6')
      CALL SLAM
      STOP

      END

*****
* Subroutine to calculate stats and controls at end of each run *
*****
* Subroutine variable definitions
* Y(J)=average time in system for J'th replication
* W(J,K)=standard work variable K for J'th replication
* S(J,K)=service time mean for center K for J'th replication
* A(J)=analytical control variate for J'th replication
* DENOM(J)=j'th denominator for calculation of external control variate
*          using Jackson network approximation
* R(J)=j'th mean interarrival time
* Number=number of replications

      SUBROUTINE OTPUT

      INCLUDE 'PARAM.INC'
      COMMON/SCOM1/ATRB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, II, MFA,
      1MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
      2SSL(MEQT),TNEXT, TNOW, XX(MMXXV)
      DOUBLE PRECISION Y(1000), W(1000,3), S(1000,3)
      DOUBLE PRECISION A(1000), R(1000)
      DOUBLE PRECISION DENOM(3), SA(1000)
      INTEGER matOpen, mxCreateFull, matClose
      INTEGER mxGetPr, MatPut Matrix
      INTEGER b, c, FP, STAT
      INTEGER J, K, NUMBER
      NUMBER = 1000
```

```

* Calculate average time in system for each replication

      Y(NNRUN) = CCAVG(5)

* Calculate standardized work variables and service time means

      DO 10 J = 1, 3
          W(NNRUN,J) = XX(J+13)/SQRT(CCNUM(J+5))
          S(NNRUN,J) = CCAVG(J)
      10 CONTINUE

* Calculate mean interarrival time and standardized arrival variable

      SA(NNRUN) = XX(17)/SQRT(CCNUM(9))
      R(NNRUN) = CCAVG(4)

* Calculate analytical control variate

      DO 20 K = 1, 3
          DENOM(K) = (1.0/S(NNRUN,K)) - (1.0/R(NNRUN))
      20 CONTINUE

      A(NNRUN) = (1.0/DENOM(1)) + (1.0/DENOM(2)) + (1.0/DENOM(3))

* Output data in MATLAB format

      IF (NNRUN .GE. NUMBER) THEN

          FP = matOpen('onethouw.mat', 'w')
          b= mxCreateFull(NUMBER,1,0)
          call mxCopyReal8ToPtr(Y, mxGetPr(b), NUMBER)
          call mxSetName(b, 'sojourn')
          stat = matPutMatrix(fp, b)

          c= mxCreateFull(NUMBER,3,0)
          call mxCopyReal8ToPtr(W, mxGetPr(c), 3*NUMBER)
          call mxSetName(c, 'stdwork')
          stat = matPutMatrix(fp, c)
          call mxFreeMatrix(c)

          call mxCopyReal8ToPtr(SA, mxGetPr(b), NUMBER)
          call mxSetName(b, 'stdarrive')
          stat = matPutMatrix(fp, b)

          call mxCopyReal8ToPtr(A, mxGetPr(b), NUMBER)
          call mxSetName(b, 'analytical')
          stat = matPutMatrix(fp, b)
          stat = matClose(fp)
          call mxFreeMatrix(b)

      END IF

      RETURN
      END

```

### A.3 SLAM II Source Code for Model M<sub>3</sub>

```

;*****
; XX(I) Variable definitions
;*****
; XX(1)=exponential service time one distribution mean
; XX(2)=exponential service time two distribution mean
; XX(3)=exponential service time three distribution mean
;*****
; Atrib definitions
;*****
; Atrib(1)=arrival time
; Atrib(2)=center one service time
; Atrib(3)=center two service time
; Atrib(4)=center three service time
; Atrib(5)=sojourn time
;*****

GEN,IRISH,MICKEY D TEST MODEL,10/14/1995,1000,Y,N,Y/Y,N,N/1,132;
LIMITS,3,5,200;
INTLC,XX(1)=15.0,XX(2)=15.5,XX(3)=15.0;
INTLC,XX(25)=0;
NETWORK;
;
      CREATE,EXPON(20,1),,1,1000,1;
      ACTIVITY;
;
; *Assign service times*
;
ASN1  ASSIGN,ATRIB(2)=EXPON(XX(1),2),ATRIB(3)=EXPON(XX(2),3),
      ATRIB(4)=EXPON(XX(3),4);
      ACTIVITY,,ORDR;
;
ORDR  QUEUE(1),,,;                               Service center one
      ACTIVITY(1)/1,ATRIB(2),,PAY;
;
PAY   QUEUE(2),,,;                               Service center two
      ACTIVITY(1)/2,ATRIB(3),,PU;
;
PU    QUEUE(3),,,;                               Service center three
      ACTIVITY(1)/3,ATRIB(4),,ASN2;
;
; * Calculate sojourn time and count customers*
;
ASN2  ASSIGN,ATRIB(5)=TNOW-ATRIB(1),XX(25)=XX(25) +1;
      ACTIVITY,,XX(25).LE.300,TERM;           Warm-up period
      ACTIVITY,,XX(25).GT.300,COL1;
;
COL1  COLCT(1),ATRIB(5),SOJOURN,,1;
      ACTIVITY,,TERM;
;
TERM  TERMINATE;
      END;
FIN;

```



#### A.4 FORTRAN Subroutine for Model $M_3$

```
*****
*   Main program                               *
*****

      PROGRAM MAIN
      DIMENSION NSET(5000)
      INCLUDE 'PARAM.INC'
      COMMON/SCOM1/ATTRIB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, II, MFA,
1MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
2SSL(MEQT),TNEXT, TNOW, XX(MMXXV)
      COMMON QSET(5000)
      EQUIVALENCE (NSET(1),QSET(1))
      NNSET=5000
      NCRDR=5
      NPRNT=6
      NTAPE=7
      OPEN(UNIT=NCRDR,FILE='fort.5')
      OPEN(UNIT=NPRNT,FILE='fort.6')
      CALL SLAM

      STOP

      END

*****
* Subroutine to calculate stats and controls at end of each run *
*****

* Subroutine variable definitions
*   Y(J)=average time in system for J'th replication
*   Number=number of replications

      SUBROUTINE OTPUT

      INCLUDE 'PARAM.INC'
      COMMON/SCOM1/ATTRIB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, II, MFA,
1MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
2SSL(MEQT),TNEXT, TNOW, XX(MMXXV)

      DOUBLE PRECISION Y(1000)
      INTEGER matOpen, mxCreateFull, matClose
      INTEGER mxGetPr, MatPut Matrix
      INTEGER b, FP, STAT
      INTEGER NUMBER

      NUMBER = 1000

* Calculate average time in system for each replication

      Y(NNRUN) = CCAVG(1)
```

\* Output data in MATLAB format

IF (NNRUN .GE. NUMBER) THEN

```
FP = matOpen('extthou.mat', 'w')
b= mxCreateFull(NUMBER,1,0)
call mxCopyReal8ToPtr(Y, mxGetPr(b), NUMBER)
call mxSetName(b, 'extcontrol')
stat = matPutMatrix(fp, b)
stat = matClose(fp)
call mxFreeMatrix(b)
```

END IF

RETURN

END

## A.5 SLAM II Source Code for Models $Q_1$ and $Q_C$

```

;*****
; XX(I) Variable definitions
;*****
; XX(1)=event counter
; XX(2)=time when first statistics are gathered
; XX(3)=standardized work variable one
; XX(4)=sum of standardized work variable one
; XX(5)=standardized work variable two
; XX(6)=sum of standardized work variable two
; XX(7)=standardized work variable three
; XX(8)=sum of standardized work variable three
; XX(9)=standardized work variable four
; XX(10)=sum of standardized work variable four
; XX(11)=standardized work variable five
; XX(12)=sum of standardized work variable five
; XX(13)=standardized work variable six
; XX(14)=sum of standardized work variable six
; XX(15)=customer capacity for subnetwork (=25 for Q-1)
; XX(20)= service time one distribution beta parameter
; XX(21)= service time one distribution alpha parameter
; XX(22)= service time one distribution mean
; XX(23)= service time one distribution standard deviation
; XX(24)= service time three distribution beta parameter
; XX(25)= service time three distribution alpha parameter
; XX(26)= service time three distribution mean
; XX(27)= service time three distribution standard deviation
; XX(28)= service time four distribution beta parameter
; XX(29)= service time four distribution alpha parameter
; XX(30)= service time four distribution mean
; XX(31)= service time four distribution standard deviation
; XX(32)= service time five distribution beta parameter
; XX(33)= service time five distribution alpha parameter
; XX(34)= service time five distribution mean
; XX(35)= service time five distribution standard deviation
; XX(36)= service time six distribution beta parameter
; XX(37)= service time six distribution alpha parameter
; XX(38)= service time six distribution mean
; XX(39)= service time six distribution standard deviation
;*****
; Atrib definitions
;*****
; Atrib(1)=time leaving center one
; Atrib(2)=center two service time
; Atrib(3)=center three service time
; ATRIB(4)=center four service time
; Atrib(5)=center five service time
; Atrib(6)=center six service time
; Atrib(7)=center one service time
; Atrib(8)=sojourn time
; Atrib(9)=routing random draw from center two
; Atrib(10)=initial starting position random draw
;*****

GEN, THOMAS H. IRISH, COMPUTER, 11/28/1995, 1000, Y, N, Y/Y, N, N/1, 132;
LIMITS, 8, 10, 25;

```

```

INTLC,XX(1)=0,XX(2)=0,XX(3)=0,XX(4)=0,XX(5)=0,XX(6)=0,XX(7)=0,XX(8)=0;
INTLC,XX(9)=0,XX(10)=0,XX(11)=0,XX(12)=0,XX(13)=0,XX(14)=0,XX(15)=25;
INTLC,XX(20)=1000.0,XX(21)=1.46824,XX(22)=100.0,XX(23)=69.2516;
INTLC,XX(24)=10.0,XX(25)=5.6476,XX(26)=1.39,XX(27)=.28477;
INTLC,XX(28)=10.0,XX(29)=5.6476,XX(30)=1.39,XX(31)=.28477;
INTLC,XX(32)=1000.0,XX(33)=2.61249,XX(34)=12.5,XX(35)=5.14217;
INTLC,XX(36)=1000.0,XX(37)=2.61249,XX(38)=12.5,XX(39)=5.14217;
NETWORK;
    RESOURCE/1,SPACE(XX(15)),7,8;
;
    CREATE,.01,,,25,1;
    ACTIVITY;
;
; *Send customers to initial service centers*
;
STRT  ASSIGN,TRIB(10)=UNFRM(0,1,8),1;
      ACTIVITY,,TRIB(10).LE..708,ASNA;
      ACTIVITY,,TRIB(10).GT..708.AND.NNRSC(1).LT.1,CAP1;
      ACTIVITY,,TRIB(10).GT..708.AND.NNRSC(1).GT.0,CAP2;
;
CAP2  AWAIT(7),SPACE,,1;                                (Center 2 in Q-c)
      ACTIVITY,,TRIB(10).GT..708.AND.TRIB(10).LE..877,TWOS;
      ACTIVITY,,TRIB(10).GT..877.AND.TRIB(10).LE..9078,SRV4;
      ACTIVITY,,TRIB(10).GT..9078.AND.TRIB(10).LE..9386,SRV4;
      ACTIVITY,,TRIB(10).GT..9386.AND.TRIB(10).LE..9693,SRV5;
      ACTIVITY,,TRIB(10).GT..9693,SRV6;
;
ASNA  ASSIGN,TRIB(7)=WEIBL(XX(20),XX(21),1),1; Service time 1
      ACTIVITY,,,Q1;
;
Q1    QUEUE(1),,,,;                                       Service center 1
      ACTIVITY(25)/1,TRIB(7);
ASN1  ASSIGN,XX(1)=XX(1)+1,TRIB(1)=TNOW,1;               Count customers
      ACTIVITY,,XX(1).GE.2000,TERM;                     End replication
      ACTIVITY,,XX(1).EQ.500;                             Start stats
      ACTIVITY,,XX(1).GT.500,WRK1;                       Collect stats
      ACTIVITY,,XX(1).LT.500,CAP1;                       Warm-up period
CLK1  ASSIGN,XX(2)=TNOW,1;                               Record stat start time
      ACTIVITY;
;
; *Calc std work cv and service time #1*
;
WRK1  ASSIGN,XX(3)=TRIB(7)/XX(23)-XX(22)/XX(23),XX(4)=XX(4) + XX(3),1;
      ACTIVITY;
WK1   COLCT(7),XX(4),WORK ONE,,1;
      ACTIVITY;
SVC1  COLCT(13),TRIB(7),SERVICE 1,,1;
      ACTIVITY,,,CAP1;
;
TWOS  ASSIGN,TRIB(2)=EXPON(1,2),1;                       Service time 2(3)
      ACTIVITY;
Q2    QUEUE(2),,,,;                                       Service center 2(3)
      ACTIVITY(1)/2,TRIB(2);
;
; *Draw routing rv, and count event*
;
ASN2  ASSIGN,XX(1)=XX(1)+1,TRIB(9)=UNFRM(0,1,10),1;
      ACTIVITY,,XX(1).GE.2000,TERM;                     End replication

```

	ACTIVITY,,XX(1).EQ.500;	Start stats
	ACTIVITY,,XX(1).GT.500,WRK2;	Collect stats
	ACTIVITY,,XX(1).LT.500,PROB;	Warm-up period
CLK2	ASSIGN,XX(2)=TNOW,1;	Record stat start time
	ACTIVITY;	
	;	
	; *Calc std work cv and service time #2(3)*	
	;	
WRK2	ASSIGN,XX(5)=ATRIB(2)-1,XX(6)=XX(6) + XX(5),1;	
	ACTIVITY;	
WKC2	COLCT(8),XX(6),WORK TWO,,1;	
	ACTIVITY;	
SVC2	COLCT(14),ATRIB(2),SERVICE 2,,1;	
	ACTIVITY,,,TWOS;	
	;	
	; *Branch to next service center*	
	;	
PROB	GOON,1;	
	ACTIVITY,,ATRIB(9).LE..2 .AND. XX(1).LT.500;	
	ACTIVITY,,ATRIB(9).LE..2 .AND. XX(1).GE.500,RTE1;	
	ACTIVITY,,ATRIB(9).GT..2 .AND. ATRIB(9).LE..56,SRV3;	
	ACTIVITY,,ATRIB(9).GT..56 .AND. ATRIB(9).LE..92,SRV4;	
	ACTIVITY,,ATRIB(9).GT..92 .AND. ATRIB(9).LE..96,SRV5;	
	ACTIVITY,,ATRIB(9).GT..96,SRV6;	
FRE1	FREE,SPACE,1;	
	ACTIVITY,,,ASNA;	
RTE1	COLCT(19),ALL,ROUTE 1,,1;	Collect routing #1 info
	ACTIVITY;	
FRE2	FREE,SPACE,1;	
	ACTIVITY,,,ASN7;	
	;	
	; *Assign service times for centers 4-6 (5-7) and calc routing info*	
	;	
SRV3	ASSIGN,ATRIB(3)=WEIBL(XX(24),XX(25),3),1;	
	ACTIVITY,,XX(1).LT.500,Q3;	
	ACTIVITY;	
RTE3	COLCT(20),ALL,ROUTE3,,1;	
	ACTIVITY,,,Q3;	
SRV4	ASSIGN,ATRIB(4)=WEIBL(XX(28),XX(29),4),1;	
	ACTIVITY,,XX(1).LT.500,Q4;	
	ACTIVITY;	
RTE4	COLCT(21),ALL,ROUTE 4,,1;	
	ACTIVITY,,,Q4;	
SRV5	ASSIGN,ATRIB(5)=WEIBL(XX(32),XX(33),5),1;	
	ACTIVITY,,XX(1).LT.500,Q5;	
	ACTIVITY;	
RTE5	COLCT(22),ALL,ROUTE 5,,1;	
	ACTIVITY,,,Q5;	
SRV6	ASSIGN,ATRIB(6)=WEIBL(XX(36),XX(37),6),1;	
	ACTIVITY,,XX(1).LT.500,Q6;	
	ACTIVITY;	
RTE6	COLCT(23),ALL,ROUTE 6,,1;	
	ACTIVITY,,,Q6;	
	;	
Q3	QUEUE(3),,,;	Service center 3(4)
	ACTIVITY(1)/3,ATRIB(3);	
ASN3	ASSIGN,XX(1)=XX(1)+1,1;	Count event
	ACTIVITY,,XX(1).GE.2000,TERM;	End replication

	ACTIVITY,,XX(1).EQ.500;	Start stats
	ACTIVITY,,XX(1).GT.500,WRK3;	Collect stats
	ACTIVITY,,XX(1).LT.500,TWOS;	Warm-up period
CLK3	ASSIGN,XX(2)=TNOW,1;	Record stat start time
	ACTIVITY;	
	;	
	; *Collect std work and service time #3(4)	
	;	
WRK3	ASSIGN,XX(7)=ATRIB(3)/XX(27)-XX(26)/XX(27),XX(8)=XX(8) + XX(7),1;	
	ACTIVITY;	
WKC3	COLCT(9),XX(8),WORK THREE,,1;	
	ACTIVITY;	
SVC3	COLCT(15),ATRIB(3),SERVICE 3,,1;	
	ACTIVITY,,,TWOS;	
	;	
Q4	QUEUE(4),,,,;	Service center 4(5)
	ACTIVITY(1)/4,ATRIB(4);	
ASN4	ASSIGN,XX(1)=XX(1)+1;	Count event
	ACTIVITY,,XX(1).GE.2000,TERM;	End replication
	ACTIVITY,,XX(1).EQ.500;	Start stats
	ACTIVITY,,XX(1).GT.500,WRK4;	Collect stats
	ACTIVITY,,XX(1).LT.500,TWOS;	Warm-up period
CLK4	ASSIGN,XX(2)=TNOW,1;	Record stats start time
	ACTIVITY;	
	;	
	; *Calc and collect std work and service time #4(5)*	
	;	
WRK4	ASSIGN,XX(9)=ATRIB(4)/XX(31)-XX(30)/XX(31),XX(10)=XX(10)+XX(9),1;	
	ACTIVITY;	
WKC4	COLCT(10),XX(10),WORK FOUR,,1;	
	ACTIVITY;	
SVC4	COLCT(16),ATRIB(4),SERVICE 4,,1;	
	ACTIVITY,,,TWOS;	
	;	
Q5	QUEUE(5),,,,;	Service center 5(6)
	ACTIVITY(1)/5,ATRIB(5);	
ASN5	ASSIGN,XX(1)=XX(1)+1;	Count event
	ACTIVITY,,XX(1).GE.2000,TERM;	End replication
	ACTIVITY,,XX(1).EQ.500;	Start stats
	ACTIVITY,,XX(1).GT.500,WRK5;	Collect stats
	ACTIVITY,,XX(1).LT.500,TWOS;	Warm-up period
CLK5	ASSIGN,XX(2)=TNOW,1;	Record stat start time
	ACTIVITY;	
	;	
	; *Calc and collect std work and service time #5(6)*	
	;	
WRK5	ASSIGN,XX(11)=ATRIB(5)/XX(35)-XX(34)/XX(35), XX(12)=XX(12)+XX(11),1;	
	ACTIVITY;	
WKC5	COLCT(11),XX(12),WORK FIVE,,1;	
	ACTIVITY;	
SVC5	COLCT(17),ATRIB(5),SERVICE 5,,1;	
	ACTIVITY,,,TWOS;	
	;	
Q6	QUEUE(6),,,,;	Service center 6(7)
	ACTIVITY(1)/6,ATRIB(6);	
ASN6	ASSIGN,XX(1)=XX(1)+1;	Count event
	ACTIVITY,,XX(1).GE.2000,TERM;	End replication

	ACTIVITY,,XX(1).EQ.500;	Start stats
	ACTIVITY,,XX(1).GT.500,WRK6;	Collect stats
	ACTIVITY,,XX(1).LT.500,TWOS;	Warm-up period
CLK6	ASSIGN,XX(2)=TNOW,1;	Record stat start time
	ACTIVITY;	
	;	
	; *Calc and collect std work and service time #6(7)*	
WRK6	ASSIGN,XX(13)=ATTRIB(6)/XX(39)-XX(38)/XX(39),	
	XX(14)=XX(14)+XX(13),1;	
	ACTIVITY;	
WKC6	COLCT(12),XX(14),WORK SIX,,1;	
	ACTIVITY;	
SVC6	COLCT(18),ATTRIB(6),SERVICE 6,,1;	
	ACTIVITY,,,TWOS;	
	;	
ASN7	ASSIGN,ATTRIB(8)=TNOW-ATTRIB(1),1;	
	ACTIVITY;	
COL1	COLCT(1),ATTRIB(8),RESPONSE TIME,,1;	Sojourn time
	ACTIVITY,,,ASNA;	
	;	
TERM	TERMINATE,1;	
	;	
CAP1	AWAIT(8),SPACE,,1;	(Service center 2 in Q-c)
	ACTIVITY,,,TWOS;	
	END;	
FIN;		

## A.6 FORTRAN Subroutine for Models $Q_1$ and $Q_c$

```
*****
*   Main program                               *
*****

      PROGRAM MAIN
      DIMENSION NSET(5000)
      INCLUDE 'PARAM.INC'
      COMMON/SCOM1/ATRB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, II, MFA,
1MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
2SSL(MEQT),TNEXT, TNOW, XX(MMXXV)
      COMMON QSET(5000)
      EQUIVALENCE (NSET(1),QSET(1))
      NNSET=5000
      NCRDR=5
      NPRNT=6
      NTAPE=7
      OPEN(UNIT=NCRDR,FILE='fort.5')
      OPEN(UNIT=NPRNT,FILE='fort.6')
      OPEN(10, FILE='analytical.in',STATUS='UNKNOWN')
      CALL SLAM
      CLOSE(10)
      STOP
      END

*****
* Subroutine to calculate stats and controls at end of each run *
*****
* Subroutine variable definitions
* Y(NNRUN) = average return time for each replication (response)
* WAIT(NNRUN,J) = average wait at J'th station for each replication
* UTILIZ(NNRUN) = average utilization of station 2 (CPU)
* W(NNRUN,J) = standardized work variable for J'th station
* R(NNRUN,J) = standardized routing variable for J'th route
* PROB(J,K) = empirical transition probability matrix
* S(NNRUN,J) = average service time for J'th station
* DENOM(J) = denominator for J'th standardized routing variable
* P(J) = actual routing probability
* NUMBER = number of replications
* CUST = number of customers in network
* STATION = number of stations in network

      SUBROUTINE OPUT

      INCLUDE 'PARAM.INC'
      COMMON/SCOM1/ATRB(MATRB), DD(MEQT), DDL(MEQT), DTNOW, II, MFA,
1MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
2SSL(MEQT),TNEXT, TNOW, XX(MMXXV)
      DOUBLE PRECISION Y(1000), W(1000,6), R(1000,5), WAIT(1000,5)
      DOUBLE PRECISION UTILIZ(1000), DENOM(5), P(5)
      REAL PROB(6,6), S(1000,6)
      INTEGER matOpen, mxCreateFull, matClose
      INTEGER mxGetPr, MatPut Matrix
      INTEGER b, c, d, FP, STAT
      INTEGER J, K, NUMBER, CH(6)
      INTEGER CUST, STATION
```



```

DATA CH / 25,1,1,1,1,1 /
DATA (PROB(1,J), J=1,6) / 0.0, 1.0, 0.0, 0.0, 0.0, 0.0 /
+ (PROB(2,J), J=1,6) / 0., 0., 0., 0., 0., 0. /
+ (PROB(3,J), J=1,6) / 0., 1., 0., 0., 0., 0. /
+ (PROB(4,J), J=1,6) / 0., 1., 0., 0., 0., 0. /
+ (PROB(5,J), J=1,6) / 0., 1., 0., 0., 0., 0. /
+ (PROB(6,J), J=1,6) / 0., 1., 0., 0., 0., 0. /

P(1) = .2
P(2) = .36
P(3) = .36
P(4) = .04
P(5) = .04

NUMBER = 1000
CUST = 25
STATION = 6

*****
* Calculate system response characteristics *
*****

Y(NNRUN) = CCAVG(1)
UTILIZ(NNRUN) = (CCAVG(14)*CCNUM(14))/(TNOW-XX(2))

*****
* Calculate internal control variates *
*****

* Calculate standardized work variables

DO 10 J = 1, 6
    W(NNRUN,J) = XX(2*J+2)/SQRT(CCNUM(J+6))
10 CONTINUE

* Calculate standardized routing variables

DENOM(1) = SQRT(CCNUM(8)*(1-P(1))*P(1))
R(NNRUN,1) = (CCNUM(7)-CCNUM(8)*P(1))/DENOM(1)
DO 20 J=2, 5
    DENOM(J) = SQRT(CCNUM(8)*(1-P(J))*P(J))
    R(NNRUN,J) = (CCNUM(J+7)-CCNUM(8)*P(J))/DENOM(J)
20 CONTINUE

* Output response data and internal cv's in MATLAB format

IF (NNRUN .GE. NUMBER) THEN

    FP = matOpen('model.mat', 'w')

    b= mxCreateFull(NUMBER,1,0)

    call mxCopyReal8ToPtr(Y, mxGetPr(b), NUMBER)
    call mxSetName(b, 'sojourn')
    stat = matPutMatrix(fp, b)
    call mxCopyReal8ToPtr(UTILIZ, mxGetPr(b), NUMBER)
    call mxSetName(b, 'utiliz')

```

```

stat = matPutMatrix(fp, b)
call mxFreeMatrix(b)

c= mxCreateFull(NUMBER,6,0)

call mxCopyReal8ToPtr(W, mxGetPr(c), 6*NUMBER)
call mxSetName(c, 'stdwork')
stat = matPutMatrix(fp, c)
call mxFreeMatrix(c)

d= mxCreateFull(NUMBER,5,0)

call mxCopyReal8ToPtr(R, mxGetPr(d), 5*NUMBER)
call mxSetName(d, 'stdroute')
stat = matPutMatrix(fp, d)
call mxCopyReal8ToPtr(WAIT, mxGetPr(d), 5*NUMBER)
call mxSetName(d, 'wait')
stat = matPutMatrix(fp, d)
call mxFreeMatrix(d)

stat = matClose(fp)

END IF

*****
* Calculate and format output for analytical control program (ForQue) *
*****

* Calculate average service times and actual routing proportions

DO 30 J=1, 6
    S(NNRUN,J) = CCAVG(J+12)
30 CONTINUE
K = 2
PROB(K,1) = CCNUM(19)/CCNUM(8)
DO 40 J=3, 6
    PROB(K,J) = CCNUM(J+17)/CCNUM(8)
40 CONTINUE

* Output data for analytical program for each replication

WRITE (10,100) CUST, STATION
WRITE (10,200)
WRITE (10,500) (S(NNRUN,J), J=1,6)
WRITE (10,300) (CH(J), J=1,6)
WRITE (10,200)
WRITE (10,400) ((PROB(J,K), K=1,6), J=1,6)
WRITE (10,200)

100 FORMAT (2I3)
200 FORMAT (/)
300 FORMAT (6I3)
400 FORMAT (6(6F9.5/))
500 FORMAT (6F9.5)
RETURN
END

```

## A.7 SLAM II Source Code for Model Q<sub>2</sub>

```

;*****
; XX(I) Variable definitions
;*****
; XX(1)=event counter
; XX(2)=time when first statistics are gathered
;*****
; Atrib definitions
;*****
; Atrib(1)=time leaving center one
; Atrib(2)=center two service time
; Atrib(3)=center three service time
; Atrib(4)=center four service time
; ATRIB(5)=center five service time
; ATRIB(6)=center six service time
; ATRIB(7)=center one service time
; ATRIB(8)=sojourn time
; ATRIB(9)=routing random draw from center two
; ATRIB(10)=initial starting position random draw
;*****

GEN,THOMAS H. IRISH,COMPUTER,11/28/1995,1000,Y,N,Y/Y,N,N/1,132;
LIMITS,6,10,25;
INTLC,XX(1)=0,XX(2)=0;
NETWORK;
;
    CREATE,0,,,25,1;
    ACTIVITY;
;
; * Send customers to initial service center*
;
STRT  ASSIGN,ATrib(10)=UNFRM(0,1,8),1;
      ACTIVITY,,ATrib(10).LE..708,ASNA;
      ACTIVITY,,ATrib(10).GT..708.AND.ATrib(10).LE..877,TWOS;
      ACTIVITY,,ATrib(10).GT..877.AND.ATrib(10).LE..9078,SRV4;
      ACTIVITY,,ATrib(10).GT..9078.AND.ATrib(10).LE..9386,SRV4;
      ACTIVITY,,ATrib(10).GT..9386.AND.ATrib(10).LE..9693,SRV5;
      ACTIVITY,,ATrib(10).GT..9693,SRV6;
;
ASNA  ASSIGN,ATrib(7)=EXPON(100,1),1;           Service time 1
      ACTIVITY,,,Q1;
;
Q1    QUEUE(1),,,,;                             Service center 1
      ACTIVITY(25)/1,ATrib(7);
ASN1  ASSIGN,XX(1)=XX(1)+1,ATrib(1)=TNOW,1;      Count event
      ACTIVITY,,XX(1).GE.2000,TERM;              End replication
      ACTIVITY,,XX(1).EQ.500;                    Start stats
      ACTIVITY,,XX(1).GT.500,ZAAH;                Collect stats
      ACTIVITY,,XX(1).LT.500,TWOS;                Warm-up period
CLK1  ASSIGN,XX(2)=TNOW,1;                       Record stat start time
      ACTIVITY;
ZAAH  GOON,1;
      ACTIVITY,,,TWOS;
;
TWOS  ASSIGN,ATrib(2)=EXPON(1,2),1;              Service time 2
      ACTIVITY;

```

```

Q2      QUEUE(2),,,,;                               Service center 2
        ACTIVITY(1)/2,ATRIB(2);
;
; *Count event and assign branching rv*
;
ASN2    ASSIGN,XX(1)=XX(1)+1,ATRIB(9)=UNFRM(0,1,10),1;
        ACTIVITY,,XX(1).GE.2000,TERM;               End replication
        ACTIVITY,,XX(1).EQ.500;                     Start stats
        ACTIVITY,,XX(1).GT.500,SVC2;                 Collect stats
        ACTIVITY,,XX(1).LT.500,PROB;                 Warm-up period
CLK2    ASSIGN,XX(2)=TNOW,1;                          Record stat start time
        ACTIVITY;
SVC2    COLCT(14),ATRIB(2),SERVICE 2,,1;
        ACTIVITY,,,TWOS;
;
PROB    GOON,1;
;
; * Branch customers to service center 3-6*
;
        ACTIVITY,,ATRIB(9).LE..2 .AND. XX(1).LT.500,ASNA;
        ACTIVITY,,ATRIB(9).LE..2 .AND. XX(1).GE.500;
        ACTIVITY,,ATRIB(9).GT..2 .AND. ATRIB(9).LE..56,SRV3;
        ACTIVITY,,ATRIB(9).GT..56 .AND. ATRIB(9).LE..92,SRV4;
        ACTIVITY,,ATRIB(9).GT..92 .AND. ATRIB(9).LE..96,SRV5;
        ACTIVITY,,ATRIB(9).GT..96,SRV6;
;
; *Assign service times*
;
SRV3    ASSIGN,ATRIB(3)=EXPON(1.39,3),1;
        ACTIVITY,,,Q3;
SRV4    ASSIGN,ATRIB(7)=EXPON(1.39,4),1;
        ACTIVITY,,,Q4;
SRV5    ASSIGN,ATRIB(3)=EXPON(12.5,5),1;
        ACTIVITY,,,Q5;
SRV6    ASSIGN,ATRIB(6)=EXPON(12.5,6),1;
        ACTIVITY,,,Q6;
;
Q3      QUEUE(3),,,,;                               Service center 3
        ACTIVITY(1)/3,ATRIB(3);
ASN3    ASSIGN,XX(1)=XX(1)+1,1;                       Count event
        ACTIVITY,,XX(1).GE.2000,TERM;               End replication
        ACTIVITY,,XX(1).EQ.500;                     Start stats
        ACTIVITY,,XX(1).GT.500,ZAAJ;                 Collect stats
        ACTIVITY,,XX(1).LT.500,TWOS;                 Warm-up period
CLK3    ASSIGN,XX(2)=TNOW,1;                          Record stat start time
        ACTIVITY;
ZAAJ    GOON,1;
        ACTIVITY,,,TWOS;
;
Q4      QUEUE(4),,,,;                               Service center 4
        ACTIVITY(1)/4,ATRIB(7);
ASN4    ASSIGN,XX(1)=XX(1)+1,1;                       Count event
        ACTIVITY,,XX(1).GE.2000,TERM;               End replication
        ACTIVITY,,XX(1).EQ.500;                     Start stats
        ACTIVITY,,XX(1).GT.500,ZAAK;                 Collect stats
        ACTIVITY,,XX(1).LT.500,TWOS;                 Warm-up period
CLK4    ASSIGN,XX(2)=TNOW,1;                          Record stat start time
        ACTIVITY;

```

ZAAK	GOON,1;	
	ACTIVITY,,,TWOS;	
;		
Q5	QUEUE(5),,,,;	Service center 5
	ACTIVITY(1)/5,TRIB(5);	
ASN5	ASSIGN,XX(1)=XX(1)+1,1;	Count event
	ACTIVITY,,,XX(1).GE.2000,TERM;	End replication
	ACTIVITY,,,XX(1).EQ.500;	Start stats
	ACTIVITY,,,XX(1).GT.500,ZAAL;	Collect stats
	ACTIVITY,,,XX(1).LT.500,TWOS;	Warm-up period
CLK5	ASSIGN,XX(2)=TNOW,1;	Record stat start time
	ACTIVITY;	
ZAAL	GOON,1;	
	ACTIVITY,,,TWOS;	
;		
Q6	QUEUE(6),,,,;	Service center 6
	ACTIVITY(1)/6,TRIB(6);	
ASN6	ASSIGN,XX(1)=XX(1)+1,1;	Count event
	ACTIVITY,,,XX(1).GE.2000,TERM;	End replication
	ACTIVITY,,,XX(1).EQ.500;	Start stats
	ACTIVITY,,,XX(1).GT.500,ZAAM;	Collect stats
	ACTIVITY,,,XX(1).LT.500,TWOS;	Warm-up period
CLK6	ASSIGN,XX(2)=TNOW,1;	Record stat start time
	ACTIVITY;	
ZAAM	GOON,1;	
	ACTIVITY,,,TWOS;	
;		
ASN7	ASSIGN,TRIB(8)=TNOW-TRIB(1),1;	
	ACTIVITY;	
COL1	COLCT(1),TRIB(8),RESPONSE TIME,,1;	Sojourn time
	ACTIVITY,,,ASNA;	
;		
TERM	TERMINATE,1;	
	END;	
FIN;		

## A.8 FORTRAN Subroutine for Model Q<sub>2</sub>

```

*****
*   Main program                               *
*****

      PROGRAM MAIN
      DIMENSION NSET(5000)
      INCLUDE 'PARAM.INC'
      COMMON/SCOM1/ATRI(MATRB), DD(MEQT), DDL(MEQT), DTNOW, II, MFA,
1MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
2SSL(MEQT),TNEXT, TNOW, XX(MMXXV)
      COMMON QSET(5000)
      EQUIVALENCE (NSET(1),QSET(1))
      NNSET=5000
      NCRDR=5
      NPRNT=6
      NTAPE=7
      OPEN(UNIT=NCRDR,FILE='fort.5')
      OPEN(UNIT=NPRNT,FILE='fort.6')
      CALL SLAM
      STOP

      END

*****
* Subroutine to calculate stats and controls at end of each run *
*****
* Subroutine variable definitions
* Y(NNRUN) = average return time for each replication (response)
* RATE(NNRUN) = average event rate
* WAIT(NNRUN,J) = average wait at J'th station for each replication
* Number = number of replications

      SUBROUTINE OTPUT

      INCLUDE 'PARAM.INC'
      COMMON/SCOM1/ATRI(MATRB), DD(MEQT), DDL(MEQT), DTNOW, II, MFA,
1MSTOP,NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(MEQT),
2SSL(MEQT),TNEXT, TNOW, XX(MMXXV)
      DOUBLE PRECISION Y(1000), WAIT(1000,5), UTILIZ(1000)
      INTEGER matOpen, mxCreateFull, matClose
      INTEGER mxGetPr, MatPut Matrix
      INTEGER b, d, FP, STAT
      INTEGER J, NUMBER

      NUMBER = 1000

*****
* Calculate system response characteristics                               *
*****

* Calculate average response time for each replication

      Y(NNRUN) = CCAVG(1)
      UTILIZ(NNRUN) = (CCAVG(14)*CCNUM(14))/(TNOW-XX(2))

```

\* Output response data in Matlab format

IF (NNRUN .GE. NUMBER) THEN

```
FP = matOpen('external.mat', 'w')
b= mxCreateFull(NUMBER,1,0)
call mxCopyReal8ToPtr(Y, mxGetPr(b), NUMBER)
call mxSetName(b, 'extcontrol')
stat = matPutMatrix(fp, b)
call mxCopyReal8ToPtr(UTILIZ, mxGetPr(b), NUMBER)
call mxSetName(b, 'extutiliz')
stat = matPutMatrix(fp, b)
call mxFreeMatrix(b)
```

```
d= mxCreateFull(NUMBER,5,0)
call mxCopyReal8ToPtr(WAIT, mxGetPr(d), 5*NUMBER)
call mxSetName(d, 'extwait')
stat = matPutMatrix(fp, d)
call mxFreeMatrix(d)
stat = matClose(fp)
```

END IF

RETURN

END

## Appendix B: MATLAB Script Files

### B.1 MATLAB Main Program for Analysis of Simulation Data

```
% Computes and compares controlled responses from SLAM data
%
% Load data
%
load model; % Internal cv's (stdwork, stdroute)
load external; % External cv's (extcontrol extutiliz)
load analytical.out -ascii; % Analytical cv's
load cntl; % t-statistics and binary matrices
load mu; % expected values of response stats
load extmu; % expected values of analytical model
%
% Set response and analytical and external control variates to be
% analyzed
%
response=[sojourn utiliz];
ext=[extcontrol extutiliz];
int=[stdwork stdroute];
[runs,number]=size(response);
for q=1:number
    resp=response(:,q);
    analytcv=analytical(:,q);
    extcontrolcv=ext(:,q);
%
% Calculate results for 10 reps/100 exps, then 20 reps/50 exps
%
    for p=1:2
        reps=10*p;
        exps=100/p;
%
% Perform analysis on {reps} replications with {exps} experiments
%
        for k=1:exps
            y=resp((k-1)*reps+1:k*reps,:);
            intcv=int((k-1)*reps+1:k*reps,:);
            analyt=analytcv((k-1)*reps+1:k*reps,:);
            extcv=extcontrolcv((k-1)*reps+1:k*reps,:);
%
% Compute uncontrolled response
%
            uncntl(k,:)=uncntlst(y,mu(q));
%
% Compute controlled responses
%
            anltcntl(k,:)=control(y,mu(q),analyt,extmu(q));
            extcntl(k,:)=control(y,mu(q),extcv,extmu(q));
            int_temp=control(y,mu(q),intcv,0.0);
            total=size(int);
            for r=1:(2^total(1,2))-1
                row=(r-1)*exps+k;
                intcntl(row,:)=int_temp(r,:);
            end %for loop-r
        end %for loop-k (all experiments of 10/20 reps)
%
%
```



```

% Calculate average results for 10/20 reps
%
% Calculate average responses
%
    if p==1
        result10(q,:)=mean(uncntl);
    %
% Calculate controlled responses
%
        red10_int=result(result10(q,:),q,p,intcntl,total(1,2));
        red10_anal=result(result10(q,:),q,p,anltcntl,1);
        red10_ext=result(result10(q,:),q,p,extcntl,1);
        if q==1
            red10=[red10_int;red10_anal;red10_ext];
        else
            red10=[red10;red10_int;red10_anal;red10_ext];
        end %if-q
        clear uncntl anltcntl extcntl intcntl int_temp
    %
    else
% Calculate averages for 20 reps
%
        result20(q,:)=mean(uncntl);
    %
% Calculate controlled responses
%
        red20_int=result(result20(q,:),q,p,intcntl,total(1,2));
        red20_anal=result(result20(q,:),q,p,anltcntl,1);
        red20_ext=result(result20(q,:),q,p,extcntl,1);
        if q==1
            red20=[red20_int;red20_anal;red20_ext];
        else
            red20=[red20;red20_int;red20_anal;red20_ext];
        end %if-q

    end %if-p (100 or 50 experiments)
end %for loop-p (10 or 20 runs)
end %for loop-q (response under study)
%
% Save results
%
save result10.out result10 -ascii
save red10.out red10 -ascii
save result20.out result20 -ascii
save red20.out red20 -ascii

```

## *B.2 MATLAB Function for Calculating Controlled Responses*

```
function z = control(y,mu,x,extmu)
load cntl;
%
% Returns controlled responses and associated statistics
% Input:
%     y = uncontrolled response
%     mu = known mean of y
%     x = control variates
%     extmu = known mean(s) of x
% Output:
%     z = [controlled mean, variance, confidence interval width,
%          coverage (0 or 1), mean square error from mu]
%
[len,wid]=size(x);
%
% Load appropriate binary matrix to compute
% all possible combinations of cv's
%
bin=binary(wid);
%
% compute z for all possible cv combinations
%
comb=(2^wid)-1;
e=ones(len,1);
x=x-extmu;
for j=1:comb
    X=[e x(:,bin(j,:))];
    beta=X\y;
    z(j,1)=beta(1,1);
    [l,w]=size(X);
    sigma=inv(X'*X);
    SSE=y'*y-beta'*X'*y;
    MSE=(1/(l-w))*SSE;
    z(j,2)=sigma(1,1)*MSE;
    z(j,3)=2*t(len-w,:)*sqrt(z(j,2));
    ul=z(j,1)+.5*z(j,3);
    ll=z(j,1)-.5*z(j,3);
    z(j,4)=mu<=ul & mu>=ll;
    z(j,5)=(z(j,1)-mu)^2;
end
```

### *B.3 MATLAB Function for Computing Uncontrolled Response and Statistics*

```
function z=uncntlst(y,mu)
load cntl;
%
% Computes uncontrolled response and statistics
%   Input:
%       y = uncontrolled response
%       mu = true mean
%   Output:
%       z = [mean, variance, confidence interval width, coverage
%            (0 or 1), mean square error from true mean]
%

z(1)=mean(y);
z(2)=(std(y)^2)/length(y);
z(3)=2*t(length(y)-1,:)*sqrt(z(2));
ul=z(1)+.5*z(3);
ll=z(1)-.5*z(3);
z(4)=mu<=ul & mu>=ll;
z(5)=(z(1)-mu)^2;
```

#### *B.4 MATLAB Function to Average Statistics over all Experiments*

```
function z=result(y,q,n,x,w)
%
% Calculate average controlled responses for 100/n experiments
%
% Input:
%   y = uncontrolled responses
%   q = response statistic indicator
%   n = counter to determine number of experiments
%   x = controlled responses
%   w = number of control variates
% Output:
%   z = [q, variance, confidence interval width, coverage, MSE,
%        variance reduction percent, confidence interval percent]
%
exps=100/n;
for m=1:(2^w)-1
    z(m,1)=q;
    z(m,2)=mean(x((m-1)*exps+1:m*exps,2));
    z(m,4)=mean(x((m-1)*exps+1:m*exps,3));
    z(m,6:7)=mean(x((m-1)*exps+1:m*exps,4:5));
    z(m,3)=(1-(z(m,2)/y(1,2)))*100;
    z(m,5)=(1-(z(m,4)/y(1,3)))*100;
    z(m,8)=mean(x((m-1)*exps+1:m*exps,1));
end
```

### *B.5 MATLAB Function for Returning Appropriate Binary Matrix*

```
function z=binary(wid)
%
% Returns appropriate "binary" file for number of control variates
% (For example if number of cv's is 3, then appropriate binary
% matrix is: [0 0 1; 0 1 0; 1 0 0; 0 1 1; 1 0 1; 1 1 0; 1 1 1] )
%
% Input:
% wid = number of control variates
% Output:
% z = [binary matrix]

if wid==1
    bin=bin1;
elseif wid==2
    bin=bin2;
elseif wid==3
    bin=bin3;
elseif wid==4
    bin=bin4;
elseif wid==5
    bin=bin5;
elseif wid==6
    bin=bin6;
if wid==7
    bin=bin7;
elseif wid==8
    bin=bin8;
elseif wid==9
    bin=bin9;
elseif wid==10
    bin=bin10;
elseif wid==11
    bin=bin11;
end
```

## Bibliography

1. Bauer, Kenneth W., Jr. and James R. Wilson. "Control-Variate Selection Criteria", *Naval Research Logistics*, 39: 307-321 (1992).
2. Bauer, Kenneth W., Jr. and James R. Wilson. "Standardized Routing Variables: A New Class of Control Variates", *Journal of Statistical Computation and Simulation*, 46: 69-78 (1993).
3. Burt, J. M. Jr., D. P. Gaver, and M. Perlas. "Simple Stochastic Networks: Some Problems and Procedures", *Naval Research Logistics Quarterly*, 17: 439-459 (1970).
4. Dietz, Dennis C. "Robustness of an Exponential Repairability model with Constrained Maintenance Resources", unpublished.
5. Dietz, Dennis C. and Richard C. Jenkins. "Analysis of Aircraft Sortie Generation Using a Fork-Join Queueing Network Model", submitted to *Naval Research Logistics Quarterly*, 31 July 1995.
6. Dietz, Dennis C. and Catherine M. Harmonosky. "Application of a Control Variate Technique to Simulation Analysis of Aircraft Sortie Generation", *IMSE Working Paper*, 89-109, June 1989.
7. *ForQue*. Computer software (unpublished). Dennis C. Dietz, 1996.
8. Gaver, D. P. and S. S. Shedler. "Control variable Methods in the Simulation of a Model of a Multiprogrammed Computer System", *Naval Research Logistics Quarterly*, 18: 435-450 (1971).
9. Jenkins, Richard J. *A Mean Value Analysis Heuristic For Analysis of Aircraft Sortie Generation*. MS thesis, AFIT/GOR/ENS/94M-07. Graduate School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 1994.
10. Lavenberg, Stephen S., Thomas L. Moeller, and Peter D. Welch. "Statistical Results on Control Variables with Application to Queueing Network Simulation". *Operations Research*, 30: 182-202 (January - February 1982).
11. Lavenberg, S. S. and M. Reiser. "Stationary Probabilities at Arrival Instants for Closed Queueing Networks with Multiple Types of Customers", *Journal of Applied Probability*, 17: 1048-1061 (1980).

12. Lavenberg, S. S. and P. D. Welch. "A Perspective on the Use of Control Variables to Increase the Efficiency of Monte Carlo Simulations", *Management Science*, 27: 322-335 (March 1981).
13. Law, Averill M. and W. David Kelton. *Simulation Modeling and Analysis* (Second Edition). New York: McGraw-Hill Book Company, 1991.
14. Nelson, Barry L. "Variance Reduction for Simulation Practitioners", *Proceedings of the 1987 Winter Simulation Conference*: 43-57 (1987).
15. Nelson, Barry L. "Control-Variate Remedies", *Operations Research*, 38: 974-992 (1990).
16. Neter, John, William Wasserman, and Michael H. Kutner. *Applied Linear Statistical Models* (Third Edition). Burr Ridge, Illinois: Richard D. Irwin, INC., 1990.
17. Pritsker, A. Alan. *Introduction to Simulation and SLAM II* (Third Edition). New York: John Wiley & Sons, 1986.
18. Ross, Sheldon M. *Introduction to Probability Models* (Fifth Edition). Boston: Academic Press, Inc., 1993.
19. Sharon, Anthony P. *The Effectiveness of Jackson Networks as Control Variates for Queueing Network Simulation*. MS thesis. Ohio State University, Columbus, OH, Winter 1986.
20. Tomick, John J. *A Comparison of Control Variates for Queueing Network Simulation*. MS thesis, AFIT/GOR/ENS/88D-22. Graduate School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1988.
21. Tomick, John J., Joseph R. Litko, and Kenneth W. Bauer. "A Comparison of Control Variates For Queueing Network Simulation", *Proceedings of the 1989 Pittsburgh Simulation Conference*, (1989).
22. Whitt, Ward A. "The Queueing Network Analyzer", *The Bell System Technical Journal*, 62: 2779-2815 (November 1983).
23. Wilson, James R. and A. Alan B. Pritsker. "Variance Reduction in Queueing Simulation Using Generalized Concomitant Variables", *Journal of Statistical Computation and Simulation* 19: 129-153 (1984).

## *Vita*

Captain Thomas H. Irish [REDACTED]

He graduated from high school in Peoria, Illinois in 1973 and enlisted in the United States Air Force in December 1976. Captain Irish separated from the Air Force in 1980 to pursue his education. He graduated from California State University, Sacramento with the degree of Bachelor of Arts in Mathematics in 1985. Captain Irish was a distinguished graduate of the United States Air Force Officer Training School and was commissioned in November 1985. His first assignment was to Fairchild AFB, Washington, as 92nd Field Maintenance Squadron Assistant Maintenance Supervisor. He has since served several tours as an Aircraft Maintenance Officer and Supply Officer. Captain Irish entered the School of Engineering, Air Force Institute of Technology in August 1994.

[REDACTED]  
[REDACTED]  
[REDACTED]



REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE March 1996		3. REPORT TYPE AND DATES COVERED Master's Thesis
4. TITLE AND SUBTITLE REPLICATIVE USE OF AN EXTERNAL MODEL IN SIMULATION VARIANCE REDUCTION			5. FUNDING NUMBERS	
6. AUTHOR(S) Thomas H. Irish, Captain, USAF				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology 2750 P Street WPAFB OH 45433-5683			8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GOA/ENS/96M-02	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution unlimited			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) The use of control variates is a well-known variance reduction technique for discrete event simulation experiments. Currently, internal control variates are used almost exclusively by practitioners and researchers when using control variates. The primary objective of this study is to explore the variance reduction achieved by the replicative use of an external, analytical model to generate control variates. Performance for the analytical control variates is compared to the performance of typical internal and external control variates for both an open and a closed queueing network. Performance measures used are confidence interval width reduction, realized coverage, and estimated Mean Square Error. Results of this study indicate analytical control variates achieve comparable confidence interval width reduction with internal and external control variates. However, the analytical control variates exhibit greater levels of estimated bias. Possible causes and remedies for the observed bias are discussed and areas for future research and use of analytical control variates conclude the study.				
14. SUBJECT TERMS Simulation, variance reduction techniques, control variates, queueing networks			15. NUMBER OF PAGES 126	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

## GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to **stay within the lines** to meet **optical scanning requirements**.

**Block 1. Agency Use Only (Leave blank).**

**Block 2. Report Date.** Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

**Block 3. Type of Report and Dates Covered.** State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

**Block 4. Title and Subtitle.** A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

**Block 5. Funding Numbers.** To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C - Contract	PR - Project
G - Grant	TA - Task
PE - Program Element	WU - Work Unit Accession No.

**Block 6. Author(s).** Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

**Block 7. Performing Organization Name(s) and Address(es).** Self-explanatory.

**Block 8. Performing Organization Report Number.** Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

**Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es).** Self-explanatory.

**Block 10. Sponsoring/Monitoring Agency Report Number.** (If known)

**Block 11. Supplementary Notes.** Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of...; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

**Block 12a. Distribution/Availability Statement.** Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

**DOD** - See DoDD 5230.24, "Distribution Statements on Technical Documents."

**DOE** - See authorities.

**NASA** - See Handbook NHB 2200.2.

**NTIS** - Leave blank.

**Block 12b. Distribution Code.**

**DOD** - Leave blank.

**DOE** - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.

**NASA** - Leave blank.

**NTIS** - Leave blank.

**Block 13. Abstract.** Include a brief (*Maximum 200 words*) factual summary of the most significant information contained in the report.

**Block 14. Subject Terms.** Keywords or phrases identifying major subjects in the report.

**Block 15. Number of Pages.** Enter the total number of pages.

**Block 16. Price Code.** Enter appropriate price code (*NTIS only*).

**Blocks 17. - 19. Security Classifications.** Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

**Block 20. Limitation of Abstract.** This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.